공학박사학위논문

# Deep learning-based Abuse Detection in Healthcare Insurance with Medical Treatment Data

## 진료 내역 데이터를 활용한 딥러닝 기반의 건강보험 남용 탐지

2020 년 8 월

서울대학교 대학원

산업공학과

이 제 혁

# Deep learning-based Abuse Detection in Healthcare Insurance with Medical Treatment Data

## 진료 내역 데이터를 활용한 딥러닝 기반의 건강보험 남용 탐지

지도교수 조 성 준

이 논문을 공학박사 학위논문으로 제출함

2020 년 7 월

서울대학교 대학원

산업공학과

이 제 혁

이제혁의 공학박사 학위논문을 인준함

2020 년 7 월

위 원 장 이 재 욱 (인)

부위원장 조 성 준 (인)

위 원 이 경 식 (인)

위 원 강 필 성 (인)

위 원 고 태 훈 (인)

# Abstract

# Deep learning-based Abuse Detection in Healthcare Insurance with Medical Treatment Data

Jehyuk Lee
Department of Industrial Engineering
The Graduate School
Seoul National University

As global life expectancy increases, spending on healthcare grows in accordance in order to improve quality of life. However, due to expensive price of medical care, the bare cost of healthcare services would inevitably places great financial burden to individuals and households. In this light, many countries have devised and established their own public healthcare insurance systems to help people receive medical services at a lower price. Since reimbursements are made ex-post, unethical practices arise, exploiting the post-payment structure of the insurance system. The archetypes of such behavior are overdiagnosis, the act of manipulating patient's diseases, and overtreatments, prescribing unnecessary drugs for the patient. These abusive behaviors are considered as one of the main sources of financial loss incurred in the healthcare system. In order to detect and prevent abuse, the national healthcare insurance hires medical professionals to manually examine whether the claim filing is medically legitimate or not. However, the review process is, unquestionably, very costly and time-consuming. In order to address these limitations, data mining techniques have been employed to detect problematic claims or abusive providers showing an abnormal billing pattern. However, these cases only used coarsely grained information such as claim-level or provider-level data. This extracted information may lead to degradation of the model's performance.

In this thesis, we proposed abuse detection methods using the medical treatment data, which is the lowest level information of the healthcare insurance claim. Firstly, we propose a scoring model based on which abusive providers are detected and show that the review process with the proposed model is more efficient than that with the previous model which uses the provider-level variables as input variables. At the same time, we devise the evaluation metrics to quantify the efficiency of the review process. Secondly, we propose the method of detecting overtreatment under seasonality, which reflects more reality to the model. We propose a model embodying multiple structures specific to DRG codes selected as important for each given department. We show that the proposed method is more robust to the seasonality than the previous method. Thirdly, we propose an overtreatment detection model accounting for heterogeneous treatment between practitioners. We proposed a network-based approach through which the relationship between the diseases and treatments is considered during the overtreatment detection process. Experimental results show that the proposed method classify the treatment well which does not explicitly exist in the training set. From these works, we show that using treatment data allows modeling abuse detection at various levels: treatment, claim, and provider-level.

**Keywords**: Abuse detection, Healthcare insurance, Medical treatment, Deep learning, Data mining

**Student Number**: 2014-30327

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

As global life expectancy increases, spending on healthcare grows in accordance in order to improve quality of life. Figure 1.1 illustrates the annual expenditure on health per capita of several countries in OECD [66]. It can clearly be observed that the expenditure on healthcare is gradually increasing. In case of South Korea, healthcare expenditure per capita jumps to double between 2008 and 2018.

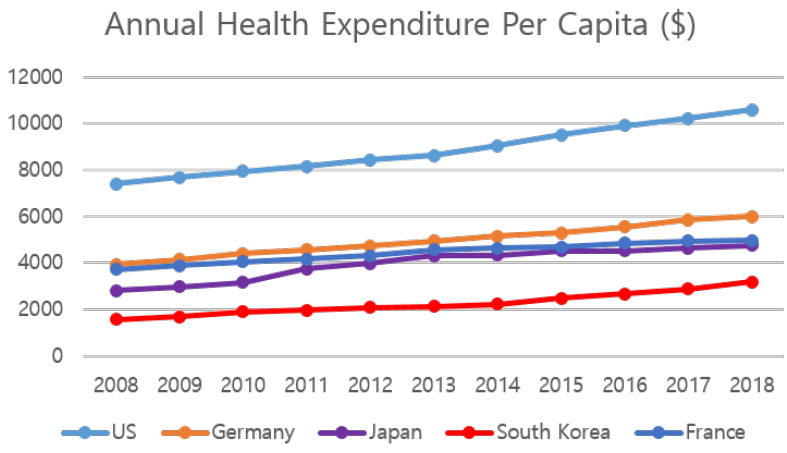


Figure 1.1: Annual expenditure on health per capita

However, medical care is quite expensive. Without a certain form of a compensation system, the bare cost of healthcare services would inevitably places great financial burden to individuals and households. In this light, many countries have devised and established their own public healthcare insurance systems to help people receive medical services at a lower price.

There exists a wild range of national healthcare insurance systems, and they differ by design from country to country. The first source of variability lies in the structure of the funding system. Canada and South Korea adopted the single-payer healthcare system, under which the government directly pays insurance fee by the means of general taxation. That is, in other words, citizens in these countries are legally obligated to pay taxes for the national health insurance. In France, compulsory contributions are made to make up the health insurance fund which are managed by non-profit organizations which are established solely for this purpose. In other countries, such as Germany or Belgium, a sickness fund is set up between employers and employees, and they make contributions to the fund. Under this system, funds are not from the government nor are they direct private payments.

On the other hand, the payment systems differ country by country. There is a variety of payment system structures including fee-for-service, bundled-payments, and global budgets. Fee-for-service refers to the payment system under which reimbursements are made for every treatment at a pre-determined unit price. This system is widely used because patients can receive quality care while providers can get reimbursed for their service. Bundled-payment, on the contrary, is a system which compensates medical expenses for the amount predetermined by the disease group to which the patient belongs to, instead of using the patient's medical history as a standard for compensation. The biggest strength of this system is that it is possible to suppress excessive treatment and increase transparency in medical expenses. Finally, global budget system estimates the total amount of medical expenses provided to the public and pays for the predetermined amount accordingly. Global budgets also has a tendency to reduce the likelihood of overdiagnosis or overtreatment. However, as compared to fee-for-service, the quality of medical care provided under bundled-payment or the global budget systems may be lower.

Given the trade-off between the quality of medical care and ethical practices, many countries run different systems simultaneously rather, than relying on a single system, in order to alleviate systematic shortcomings. France adopted all of the three systems, fee-for-service, global budget, and bundled payment system. Germany has established a modified version of global budgets and combined it with fee-for-service system. The health insurance system of South Korea, which will be the main focus of this study, takes the form of the fee-for-service system, which compensates the practitioners for their service. However, for seven disease groups, the reimbursement process takes the form of bundled-payment system, which compensates the practitioners by pre-determined payment, no matter how many treatments are provided to the patient. In other words, DRG-codes are incorporated in order to complement the limitations of the fee-for-service system with bundled payments-like apparatus. In order to broaden the scope of the system and the range of its application, the National Health Insurance Corporation of South Korea (NHIS) is implementing extensions to the bundled- payment system for patients who are not in pre-defined seven patient groups.

Since reimbursements are made ex post, unethical practices arise, exploiting the post-payment structure of the insurance system. The archetypes of such behavior are overdiagnosis, the act of manipulating patient's diseases, and overtreatments, prescribing unnecessary drugs for the patient. The loopholes in the system allow room for medical providers to prescribe excessive medical treatment or request non-existent medical treatment should they want to. Federal Bureau of Investigation (FBI) provides an extensive list of medical fraud cases by type, relevant health insurance information along with pertinent characteristic information, in the Financial Crime Report [27]. According to the report, the total fraudulent billing for health care programs amounts to be at least 3% of the total health expenditure which is estimated to be around \$2.4 trillion [27]. In addition, according to a report issued by the Korean Financial Supervisory Service, the amount of financial loss incurred by fraudulent activities was estimated to be about \$1.8 million in 2018, with further damages expected [43]. Such practices do not only increase the burden of medical expenses on the patients but also incur unnecessary social costs and expenditures. Some studies have reported that approximately 10% of medical spending is wasted due to these types of unethical practices ([22], [84]).

At this stage, let us clearly define fraud and abuse within the scope of healthcare insurance. These words appear ubiquitously in various situations, and it is difficult to disentangle the underlying meanings. Following the convention, We define fraud as a type of dishonest or intentional act which leads to unauthorized benefits for the person who commits the act or to someone else who is not entitled to the benefit [77]. On the other hand, we define abuse as a medical service or practice not consistent with the generally accepted sound fiscal practices [77]. Into the category of medical frauds fall the cases where medical service is documented and charged yet not really performed, or when a diagnosis on a patient is falsified in order to justify the unnecessary medical procedure. Abuse may lead to prescriptions that do not meet the medically stable criteria, or may result in incurring unnecessary costs by deliberately executing medically gratuitous treatments. Examples of abuse are overtreatment or improper billing practices.

Overtreatment, in particular, is considered as one of the main sources of financial loss incurred in the healthcare system. According to the report by Institute of Medicine, prescribing unnecessary services is the primary contributor to the loss incurred in the U.S. healthcare system to waste in US healthcare [59]. The report estimated that these behaviors account for approximately $210 billion out of the $750 billion loss in a year. Furthermore, a survey study, which collected survey the results from he Survey of overutilization of surveying 2106 physicians in the United States, about 20.6% of treatment is perceived as unnecessary [55]. Past literature has shown that such inappropriate or unnecessary treatments were especially conspicuous specialty care hospitals ([12], [32], [48]).

In order to detect and prevent abuse, the national healthcare insurance hires medical professionals to manually examine whether the claim filing is medically legitimate or not. However, the review process is, unquestionably, very costly and time-consuming. Moreover, there are not enough professionals to examine millions of claims. For example, in case of South Korea, there were only about 1,700 reviewers for 1.5 billion claims filed in 2016 [31]. Clearly, it is not possible to manually examine all the claims.

In response, insurance companies have resorted to an automated rule-based review system [79]. Although it can save much time and effort from reviewing all the claims manually, rule-based review system at the current level can only detect very simple abusive practices. Moreover, because this system is based on a set of predefined rules, it cannot cope with the new types of frauds and abuses rising over time.

In order to address these limitations, data mining techniques have been employed to detect abusive claims or providers showing an abnormal billing pattern ([4], [6], [30], [44], [52], [53], [65], [68], [79], [80], [82], [98], [103]). Based on these studies, insurance companies develop models detects abusive providers or problematic claims and examine relevant claims. However, these cases only used coarsely grained information such as claim-level or provider-level data. The lowest-level of information available from a claim, nonetheless, is the set of medical treatments, where patient's diseases and the set of corresponding medical activities are listed. A claim is a collection of several medical treatments, while abuse may be incurred as the result of a single or multiple medical treatments. Similarly, a provider can be represented by filed claims, while abuse may be incurred as the result of a single claim or multiple claims. So far, past literature has relied on the claim-level analysis or provider-level analysis, hence losing detailed information of each abuse in their detection models.

In this dissertation, we proposed an abuse detection methods in healthcare insurance using the medical treatment data, which is the lowest level information of the healthcare insurance claim. By using the lowest-level information, we show that it is possible to detect abuse from the healthcare insurance claims more precisely than the model with derived high-level variables. We also show that it is possible to detect abuse at various levels such as providers, claims, and treatments. First, we propose a scoring model based on which abusive providers are detected. We showed that the review process with proposed method is more efficient than with previous method, which is a scoring model with provider-level information. Second, we propose a detection model under the change of claim distribution, which reflects more reality condition. The proposed method is more robust to the change of distribution than the previous method. Third, we propose a detection model accounting for different prescription to the same patient, which reflects more reality condition. The proposed method understands the context of each entity by utilizing graph embedding method. The proposed model shows better performance than the model that does not include the context of each object. As can be seen here, we have proposed detection methods in situations similar to the real world.

This dissertation is organized as follows. In chapter 2, we proposed a neural network-based method of measuring the degree of abuse of medical service providers and selecting the abusive provider. Our model is, to our best knowledge, among the first to detect abuse in healthcare insurance using medical treatment data. In chapter 3, we propose overtreatment detection model which accounts for seasonality in claims by exploiting the concept of diagnosis-related groups, which was originally devised to classify patients. We showed that incorporating diagnosis-related groups during the claim review process helps detecting abuse better. In chapter 4, we propose an overtreatment detection model which extracts the relationship between the disease and the treatment by using graph embedding methods. Finally, we discuss the contributions and future work of this dissertation in chapter 5.

Table 1.1: Target abuse type, problem and proposed method covered in this dissertation

| **Chapter** | **Abuse unit** | **Problem** | **Proposed Methods** |
|---|---|---|---|
| Chapter 2 | Provider | Scoring the providers' degree of abuse | - Score the degree of abuse of provider with treatment data<br>- Used method: neural network with embedding layers |
| Chapter 3 | Treatment | Seasonality in the distribution of claims | - Operate the classification model by DRG code unit<br>- Used method: neural network with embedding layers |
| Chapter 4 | Treatment | Heterogeneous Treatment between Practitioners | - Modeling the disease-treatment relationship explicitly<br>- Used method: link prediction with graph embedding |

# Chapter 2

# Detection of Abusive Providers by department with Neural Network

## 2.1 Background

Abuse in healthcare refers to behaviors of providing unnecessary care to the patient. When an insurance company compensates for these unnecessary behaviors, it leads to the loss of the company. If this company is a national healthcare insurance company, it leads to an increase in premiums. In the case of countries with the single-payer healthcare system, such as South Korea, all taxpayers in the country will suffer from this loss. In other words, it can lead to a social loss in as sense that people cannot receive healthcare services at affordable prices. Due to this reason, abuse detection is an important task to solve for the healthcare insurance company, no matter if it is private or public.

In order to prevent the loss, they hire medical experts to detect these unnecessary behaviors. The problem is there are not enough experts to examine a bunch of claims. Moreover, to examine the healthcare claims, reviewers are required to know much more background knowledge than the other insurance. It means that it is more difficult to hire experts than other insurance. In order to tackle these difficulties, efforts were made to increase the efficiency of the review process. Instead of examining all claims carefully, reviewers select some problematic claims and manually review them. The objective here is set to reduce as much cost as possible by detecting as much abuse as possible with limited labor.

Now, the important issue now boils down to how these problematic claims should be selected. If a large proportion of the selected claims involves overtreatments, the reviewers can detect lots of abuse and reduce as much waste. If not, the effect of the examination would be insignificant. There have been studies that aim to detect these problematic claims using datamining techniques. These studies can be divided into two groups: detecting problematic claims and detecting abusive providers. The key assumption underlying the literature on 'detecting abusive providers' is that practitioners practice in a homogeneous pattern. This assumption would, in turn, lead to the conclusion that claims from abusive providers are more likely to include greater number of overtreatments. That is, in other words, if the reviewers can determine candidates for highly likely abusive providers and examine their claims as a priority, then there's a greater change to detect many more abuse claims in a shorter amount of time, hence being able to recover the loss induced by abuse. South Korea's HIRA screens through all the claim filings using their own scoring model to detect abusive provider candidates. The scoring model relies on datamining techniques, and the data the model learns is at the provider level.

However, previous studies do not use all of the detailed information residing in raw data. Past literature utilizes derived variables computed at the claim-level information or the provider-level. This can lead to poor performance of the model. Field experts from HIRA have expressed their aspiration to advance their existing model and suggest points of improvement. Their belief is that the primary reason of poor performance resides in the limitations in the input variables. Input variables currently in use are at the provider-level, hence incapable of accounting for different characteristics across providers. For example, there may be providers with a relatively small number of visits yet with large amounts of medical expenses, while some other providers have a relatively large number of visits with small amounts of medical expenses. Patient visits and medical expenses may vary according to the size of the provider. Failure to account for provider-wise variations may lead to degradation of the model's performance. Moreover, different diseases may be associated with different forms of abuse, yet provider-level variables cannot account for disease-wise variations either.

In this chapter, we address these issues and propose a model that scores the degree of medical abuse by provider using medical treatment data. The proposed method consists of two steps: training a neural network which scores the degree of abuse from each medical treatment, and then calculating the abuse score of each treatment by multiplying the neural network result by the claimed amount. Finally, abuse scores of the treatments are aggregated to the provider level. We define the resulting score to be the abuse score of the subject provider. We test the proposed model using in-patient claim data from six different departments in the year of 2016. Experiment results show that the proposed model is more efficient than the existing model which uses only provider-level variables. In addition, we show that the proposed model scores providers well as compared to the previous model.

The rest of the chapter is organized as follows. In section 2.2, we review the past literature on data mining methods for abnormality detection in healthcare insurance. Section 2.3 provides detailed descriptions of the proposed model. In section

2.4, we elaborate on experiment settings. We also describe the devised evaluation measures in this section. Section 2.5 reports the experiment results. Finally, section 2.6 concludes the paper.

## 2.2 Literature Review

### 2.2.1 Abnormality Detection in Healthcare Insurance with Datamining Technique

There are many studies on detecting fraud or abuse in the health insurance industry. In this subsection, we briefly survey through two major branches of health insurance abnormality detection: detecting abnormal providers, detecting abnormal claims.

**Detecting abnormal providers**

First, we briefly review several studies related to detecting abnormal providers. Here, the term 'provider' means medical service provider which provides medical service to patients such as medical institutions, general practitioners. We define abnormal providers as the providers that have different billing patterns to others. Most studies that aim to detect these providers suggest models with provider-level variables. In most cases, these variables are extracted from the raw data.

He et al. [30] applied the multi-layer perceptron (MLP) to detect abnormal General Practitioners (GPs) with sampled profile data of practicing GPs. They used 28 GP-level features that are selected by consultants. Also, the profiles were labeled on a 1-4 scale. They trained a multilayer perceptron that with this data. Also, they utilized the self-organized map (SOM) [42] with the MLP to classify the GP practice profiles.

Shan et al. [79] used the association rule mining method to make rules for detecting abnormal providers. These rules include both positive rules and negative rules. They applied the method in real claim data from Medicare Australia's Enterprise Warehouse. As a result, they extracted 215 rules and evaluated qualitatively and quantitatively. Users are willing to use this method because they can interpret the abuse though they can detect only simple abuses with these rules.

Shan et al. [80] detected abnormal providers by utilizing the local outlier factor (LOF) method which is a kind of unsupervised learning approach. They applied the method in the Australian optometrist dataset using 12 provider-level variables. They found the proposed approach outperforms domain-knowledge based methods. It means even if data is not fully labeled, the unsupervised method may be a good method to detect providers with abnormal billing patterns.

Liou et al. [53] conducted a study of detecting abnormal providers with extracted cost-related variables such as average drug cost, average diagnosis fee, or average medical expenditure per day. They trained three supervised learning models with the claim data from Taiwan's National Health Insurance using these variables. The three models were logistic regression, neural network, and classification tree. They found that the proposed model classifies abnormal providers from all providers well.

Lin et al. [52] suggested a knowledge discovery in database (KDD) approach based method that aims to detect abnormal GPs. The proposed method includes these processes. First, extract GPs' profiles from the claim databases. Here, the profile means the provider-level information such as the amount of fee, amount of prescription days, or average drug fee per case. From these variables, segment the providers using clustering methods such as SOM or PCA. Then, they described the billing patterns of each segment and provided the detailed managerial guidance that is from domain experts. They selected abnormal segments based on this guidance. Finally, the providers in such segments are considered as abnormal providers. They applied this method to the claim data from the National Health Insurance of Taiwan. The result was promising in that the model detects the abnormal providers efficiently.

Shin et al. [82] devised a scoring model that scores the provider's degree of abuse. Also, they claimed that the score from their model can be used to detect the abusive providers. The scoring model is includes following steps. First, calculate the degree of anomaly (DA) for each variable which means the deviation from the average value. Then, define the composite degree of anomaly (CDA) as a weighted average of DA and calculate CDA for each provider. The provider's CDA value is considered as the provider's degree of abuse. After the CDA value for each provider is calculated, derive the grade for degree of anomaly (GDA) by segment the CDA into several groups. In order to use these scores in detecting abusive providers, train a decision tree model using provider's profiles as input variables, and GDA value as a target variable. They applied this method to the outpatient claim data from HIRA in South Korea. They found that the proposed model is able to detect abusive providers well and easy to update.

These studies are about detecting abnormal providers using each provider's profile information. In other words, these models only use each provider's information, not the provider-provider relationship. A study conducted by Wang et al. [98] is about detecting abusive providers using the relationship. They constructed a social network of patients and providers from patients' visit sequences. For example, suppose a patient visits provider A. If the patient shows no improvement, he may visit different provider B. If then, make a directed edge from provider A to provider

B. It means if a node has high out-degree, the provider corresponding to the node can be considered suspicious. After the network is constructed, calculate the trustworthiness score for each node. Then, select suspicious providers and consider them as abnormal providers. They applied this method to both simulated and real-world claim data from National Health Insurance of Taiwan. They found that their method is effective in identifying abnormal providers. Also, they claimed that reviewers can detect abnormal providers effectively if the proposed method is used with traditional methods.

### Detecting abnormal claims

We can define abnormal claims as the claims that have different billing patterns to others or including overtreatment. Most papers that aims to detect these claims suggest models with claim-level variables. The variables are also extracted from the raw data. Yang and Hwang [103] used the process mining framework to detect abnormal claims. They define a term clinical pathway, which means frequent clinical patterns from clinical instances. If an instance deviated from the pathway, it is considered as an abnormal claim. They applied the proposed method in claim data from NHI program of Taiwan. Their experiment shows that the proposed method is more efficient than manually constructed detection models.

Ortega et al. [68] suggest a framework that detects abnormalities using the neural network. This model is not aimed to detect abnormal claims only. It aims to detect 'abnormalities'. First, they define four import entities that play important roles in healthcare insurance: medical claims, affiliates, medical professionals, and employers. Then, train neural networks for each entity that detects abnormalities. Also, the classification model and result from one entity give feedback to other models to improve the model performance. They applied the proposed framework in real claim data from private pre-paid health insurance plans(ISAPRE) in Chile. They found that the proposed model shows better performance. Also, the model shows good performance even if a new input data has quite different patterns to previous data.

Aral et al. [4] supposed a model that calculates the fraudulent risk of a claim with cross-feature analysis. The fraudulent risk is calculated from incidence matrices that are derived from a correlated variable pair. Risk metrics from both categorical features and ordinal features are calculated from the incidence matrices. They applied the proposed method to real claim data from Turkey. It shows that their approach is capable of detecting abnormalities. Moreover, another important feature of the model is the fact that it can be used in online because the inference time of this model is very short.

A framework that Bayerstadler et al. [6] devised is quite different. They try to model the claim with Bayesian multinomial latent variable. They assumed every claim is in one of following categories: 'Unperformed services'(UP), 'Unjustified services'(UJ), 'Other billing issues'(BI), and 'No irregularities'(NI). Then, a claim $i$ follows the multinomial distribution with several parameters. In order to estimate these parameters, they used the multinomial logit model and Markov Chain Monte Carlo (MCMC) sampling. They confirmed that the performance of their model showed better performance than other benchmark models.

One of the weaknesses of previous models is that they are not proper models to detect evolving frauds or abuses. In order to detect these changing abnormalities, the model needs to be re-training. However, it is difficult to know the right time to retrain, because medical claim data is different from stream data. Ngufor and

Wojtusiak [65] suggested a change point detection model with the concept drift method to solve this problem. Also, they suggested an abnormal claim detection method after detecting these change points. They applied the proposed method to simulated data and real claim data from INOVA Health System of Northern Virginia.

The approach of Kose et al. [44] suggested is quite different from previous models. They asserted that fraudulent behavior should be considered as provider-claim pair, not the claim or provider itself. They claimed that frauds are from the behaviors of multiple actor types (providers) and multiple commodities(claims). Also, they utilized the interactive machine learning approach in order to make the model adapt to changing fraud types. They found the proposed method is capable of detecting abnormalities well.

### 2.2.2 Feed-Forward Neural Network

The feed-forward neural network is a type of an artificial neural network, in which the nodes in the model do not form a cycle [105]. In other words, the information only moves from input nodes to output nodes through hidden nodes without any backward moves. It is the simplest form of the artificial neural network. If there is only a single layer of output nodes, it is called a single-layer perceptron network. If the network consists of several layers, then it is called a multilayer perceptron (MLP) network.

The main goal of the feed-forward network is to approximate a function. According to the universal approximation theorem, a feed-forward neural network comprising a single hidden layer with an activation function and a linear output layer can approximate continuous functions on the compact subset of $\mathbf{R}^{\mathbf{n}}$ ([19], [34]). That is,

Table 2.1: Previous studies about the abnormal detection in healthcare insurance

| Type of abnormality | Authors | Data mining approach | Method |
|---|---|---|---|
| Provider | He et al. [30] | Supervised | Neural network |
| | Lin et al. [52] | Unsupervised | PCA, SOM |
| | Liou et al. [53] | Supervised | Classification tree, logistic regression, neural network |
| | Shan et al. [79] | Unsupervised | Association rules |
| | Shan et al. [80] | Unsupervised | Local density based outlier detection |
| | Shin et al. [82] | Supervised | Distance based method for univariate variable, decision tree |
| | Wang et al. [98] | Unsupervised | Network analysis |
| Claim | Yang and Hwang [103] | Supervised | Process mining, association rules |
| | Ortega et al. [68] | Supervised | Neural network |
| | Aral et al. [4] | Hybrid | Distance based correlation and risk matrices |
| | Ngufor and Wojtusiak [65] | Hybrid | Change point detection, unsupervised data labeling, classification model |
| | Bayerstadler et al. [6] | Supervised | Latent variable modeling, MCMC |
| Behavior (Providers-claims) | Kose et al. [44] | Interactive | Analytic hierarchical processing(AHP), EM algorithm, data visualization |

in other words, a large MLP may represent any function given proper parameters. However, it does not guarantee that the training algorithm will be able to learn that function for sure.

In many cases, the back-propagation algorithm is used to train a neural network [78]. When the input value $\boldsymbol{x}$ generates an output value $\hat{y}$, the scalar error $E(\boldsymbol{\theta})$ is calculated, with $\boldsymbol{\theta}$ representing the set of parameters in the model. The back-propagation algorithm allows moving this information from the output layer to the input layer while computing the gradient. The network parameters are updated according to these gradients by $\Delta\boldsymbol{\theta} = -\alpha \cdot (\delta E(\theta)/\delta\boldsymbol{\theta})$, in order to minimize the error function.

Consider a $m$-layer feed-forward neural network which is fully connected. The input dimension is $n$, and the output dimension is 1. Let us define some notations as follows.

- $w_{ij}^k$: weight for perceptron $j$ in the $k$-th hidden layer for the incoming node $i$ in the $(k-1)$-th hidden layer
- $b_i^k$: bias of perceptron $i$ in the $k$-th hidden layer
- $h_i^k$: the product sum plus the bias of perceptron $i$ in the $k$-th hidden layer
- $g_h$: activation function of the hidden layers
- $g_o$: activation function of the output layers
- $o_i^k$: the output of the node $i$ in the $k$-th hidden layer
- $r_k$: number of the nodes in the $k$-th hidden layer
- $\boldsymbol{w_i^k}$: weight vector of perceptron $i$ in the $k$-th layer. $\boldsymbol{w_i^k} = \{w_{1i}^k, ..., w_{r_k i}^k\}$
- $\boldsymbol{o^k}$: output vector of $k$-th layer. $\boldsymbol{o_k} = \{o_1^k, ..., o_{r_k i}\}$

Then, the output of the neural network can be expressed as follows

$$\hat{y} = g_o(\boldsymbol{w_i^m} \cdot \boldsymbol{o^{m-1}} + b_1^m)$$

Where $h_i^k = \boldsymbol{w_i^k} \cdot \boldsymbol{o^{k-1}} + b_i^k, o_i^k = g_h(h_i^k)$, for $i = 1, 2, ..., r_k$

## 2.3 Proposed Method

This section presents a scoring model that measures provider's degree of abuse by using medical treatments. The model should give higher score to more abusive providers if the model is well trained. Then, the review process might be efficient if the reviewers only review claims from providers with high score.

At this point, we clearly define the degree of abuse. Once a provider submits a claim to the insurance company, the reviewers examine all the treatments appearing in the claim. Then, they determine whether each treatment is abused or not. If a treatment is adjudged as abuse, the amount of abuse is determined in following way: if the treatment is considered to be totally unnecessary to the patient, the abused amount equals the amount claimed; if the treatment is considered to be necessary yet excessive, then the abused amount is less than the claimed amount. The insurance company reimburse the providers for the total claimed amount, excluding the abused amount. In this paper, we define the degree of abuse of a provider as the total abused amount from whole claims that is submitted by the provider.

The proposed method consists of two steps. First, we train a model that calculates the likelihood of abuse for each medical treatment. The model is a kind of neural network that classifies whether the treatment is normal or abuse. Upon the completion of calculating the likelihood for each treatment in the test set, the result form the neural network is multiplied by the claimed amount the resulting measure of which we define as the abuse score of the subject treatment. Then, aggregate the abuse score for each treatment to the provider-level by combining scores if the treatments came from the same provider. We define the result as the abuse score of the provider. Figure 2.1 summarizes the whole framework of the provider's degree of abuse.

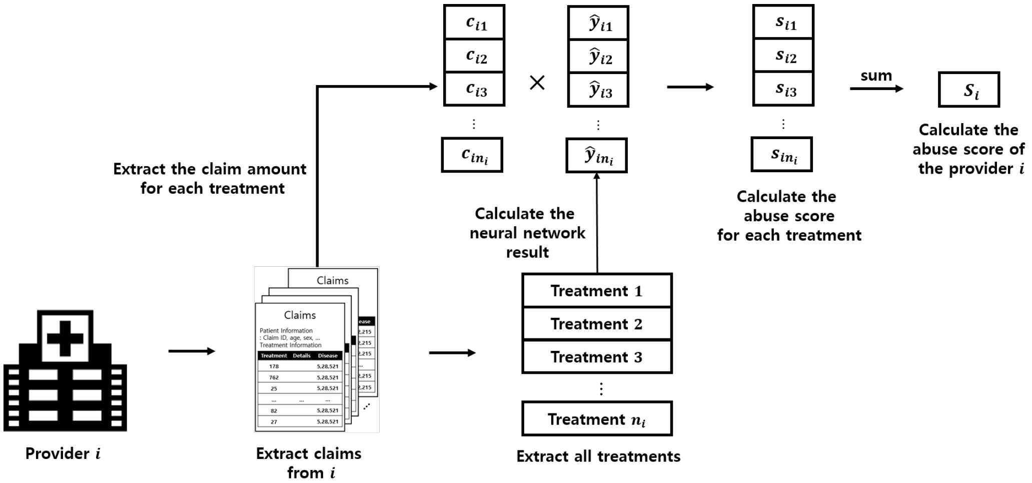


Figure 2.1: The process of scoring a provider's degree of abuse

### 2.3.1 Calculating the Likelihood of Abuse for each Treatment with Deep Neural Network

The proposed model employs a deep neural network to calculate the likelihood of abuse for each treatment. The model uses the documented information regarding to each treatment as input variables. The input variables include patient-related information, medical treatment-related information. The patient-related information includes age, gender, diseases, as well as the medical treatment-related information includes the type of operation, the unit price of medicine, or the number of medication days. The model structure is illustrated in Figure 2.2.

As illustrated in Figure 2.2, the proposed model uses both numerical variables and categorical variables. One of the most common-approaches to deal with cat-

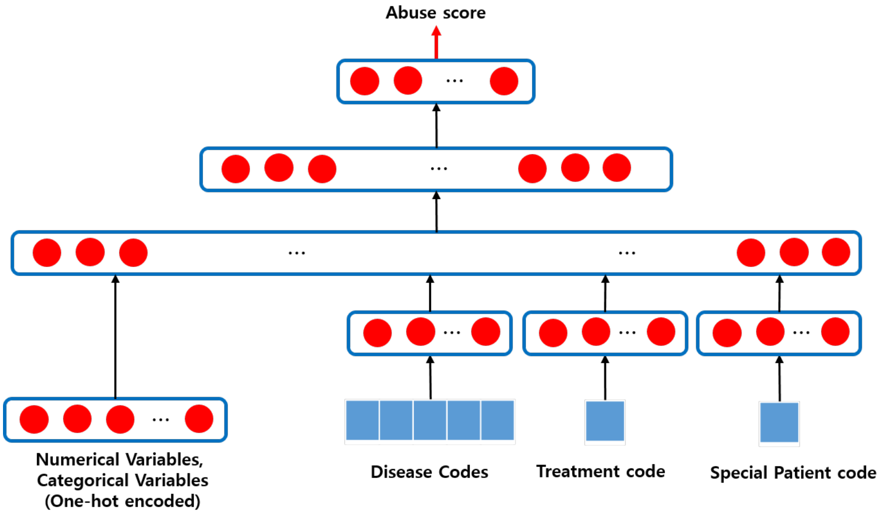


Figure 2.2: Structure of treatment scoring model

egorical variables is one-hot encoding. However, this method is undesirable if the categorical variables are of high cardinality. The data dimension will be exploded if we convert the categorical variables to numerical vectors using one-hot encoding method. Instead, we hire an embedding function to convert those variables. Our proposed model trains the embedding function as part of the training phase of the entire network. The classification error is back-propagated to embedding layers as well as hidden layers. We illustrate the training phase mathematically as follows. Suppose that we want to represent a category variable with cardinality $V$ as a $d$-dimensional vector, which is $d \ll V$. The embedding vector can be calculated as follows.

$$\boldsymbol{h} = f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{W}$$

Here, the embedding matrix $\boldsymbol{W}$ is also trained with the neural network as it minimizes the total error function during the training phase. In this case, the total loss takes the form of a binary cross entropy function, defined as follows.

$$\mathbf{L} = -\sum_{i=1}^{N}(y_i \log \hat{y_i} + (1 - y_i) \log(1 - \hat{y_i})$$

The classification error is back-propagated through the hidden layers and the embedding layers, and the parameters for both of the layers are updated as error is minimized.

In our model, we also account for a special case: the multi-valued categorical variable. In this study, the patients' disease information variable has such characteristics. When a patient visits the medical provider, his/her medical record for the visit is mostly likely to be associated with more than one disease. In this case, the disease variable has multiple values.

If the claim data includes the association relationship between disease and treatment, it would be easy to determine whether the treatment is appropriate to the patient. Unfortunately, most of the claims do not include this relationship information. Moreover, it is difficult to disentangle medical activities one by one and determine its relationship to the corresponding disease explicitly, because it may be the case that some activities are prescribed under the consideration of the potential interaction of multiple diseases. Instead, the claim includes the all diseases of the patient and all treatments details. In order to utilize treatment data in modeling, all diseases in the claim should be matched with every treatment in the claim.

Due to the lack of relationship information between disease and treatment, we average the embedding vector by disease category to represent the diseases as nu-meric vectors. The method for calculating the embedding vector for each disease code and other high-cardinality category variables are illustrated in Figure 2.3.

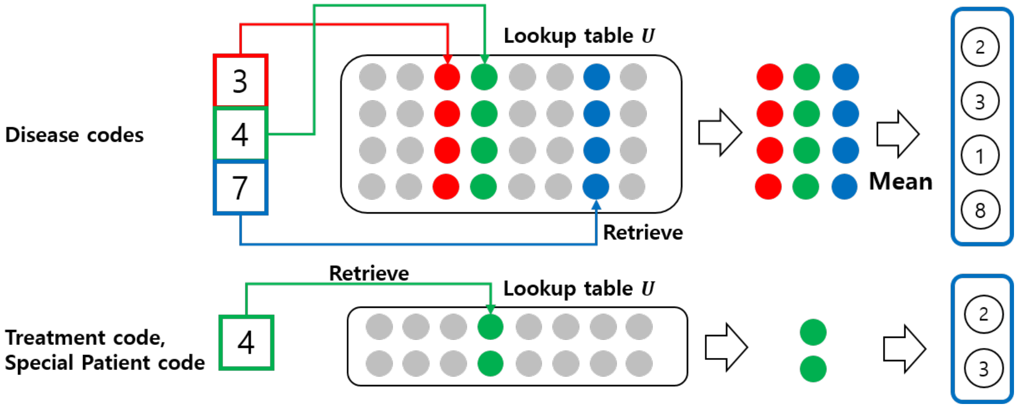


Figure 2.3: Embedding method of categorical variables with high-cardinality

### 2.3.2 Calculating the Abuse Score of the Provider

In this subsection, we describe the process of computing abuse score for each provider based on the calculated result from the neural network, which we described in subsection 2.3.1. In this subsection, we describe the process for computing the abuse score for each provider based on the results from the neural network. Suppose there are $N$ providers and with $m_1, m_2, \ldots, m_N$ claims, and $n_1, n_2, \ldots, n_N$ treatments. The amount claimed for $j$-th medical treatment by provider $i$ is represented by $c_{ij}$. The abuse likelihood of the treatment calculated by the medical treatment scoring model is represented by $\hat{y}_{ij}$. Now, the abuse score of the provider $i$ is computed in two steps: (1) the abuse score is determined for each treatment ($s_{ij}$), and then (2) the abuse scores are aggregated across treatments if they came from the same provider ($S_i$). Above two steps are summarized below.

- Calculate the abuse score of each treatment ($s_{ij}$)

$$s_{ij} = c_{ij}\hat{y}_{ij}, j = 1, 2, \ldots, n_i, i = 1, 2, \ldots, N$$

- Calculate the abuse score of each provider ($S_i$)

$$S_i = \sum_{j=1}^{n_i} s_{ij}, j = 1, 2, \ldots, n_i, i = 1, 2, \ldots, N$$

$S_i$ represents the abuse score of provider $i$, which measures the degree of abuse by the provider $i$. In order to maximize the efficiency in the reviewing process, we include the amount claimed when calculating the abuse score. Suppose there are two providers with the same number of treatments with the same abuse likelihood. Now, suppose the claimed amount for each treatment of one provider is larger than that of the other. Then, it is likely that the social cost incurred by the abuse of the former provider is larger than the latter. If reviewers can detect such abuse, the social benefit from the former is larger than the latter. This means that the reviewers can examine efficiently in a sense that they can detect abuse cases with greater social cost with smaller amount of input labor.

## 2.4 Experiments

We apply the proposed method to real-world claims data, which were submitted to HIRA in 2016. We compare the performance of the proposed model against the previous model employed by HIRA, which utilizes provider-level variables. In subsection 2.4.1, we provide the detailed descriptions of the data. Training details can be found in subsection 2.4.2. In subsection 2.4.3 and 2.4.4, we describe the evaluation measures devised for proper performance comparison among different models.

### 2.4.1 Data Description

In this dissertation, we used healthcare insurance claims submitted to the NHIS for experiments. Before we introduce our work, we provide detailed description of the filing and review process of health insurance claims.

The majority of South Korea citizens are covered by a uniform health insurance policy administered by NHIS. When a patient receives medical care from a medical service provider, the provider submits the claim for reimbursement of the amount determined by the fee-for-service policy. Then, the patient only pays for the remaining amount. Although the government strictly regulates the reimbursement process, abuse cannot be perfectly prevented, which eventually causes waste of the healthcare budget. The Health Insurance Review and Assessment (HIRA) is an institution dedicated for the detection of such abuses by investigating medical claims and auditing medical institutions. Once a medical provider submits a claim to HIRA, the reviewers examine the claim and determine whether the claim is suspicious of abuse or fraud. Then, HIRA submits the examination results to the NHIS where the reimbursements are made in accordance with the results to the subject provider. The reimbursement process is represented in Figure 2.4.

HIRA takes several steps when examining claims. When a claim is filed to HIRA, an automatic system initially checks whether there is an error in the basic information of the claim. This process is referred to as the automatic checkup process. Then, the claim goes under the process called an electronic review. In this step, a model

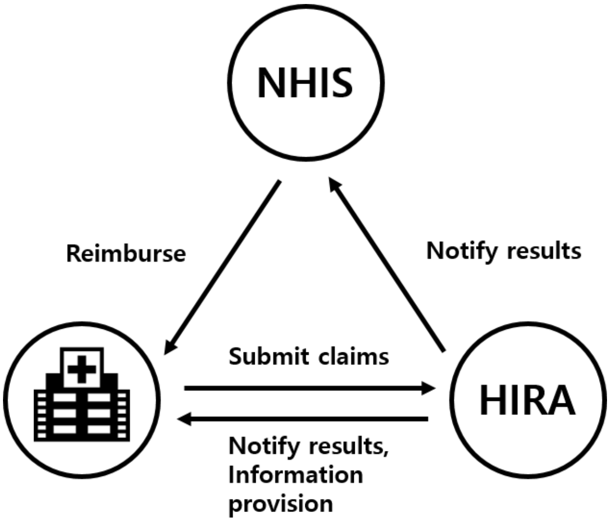


Figure 2.4: The reimbursement process of NHIS

detects whether the claim is abuse claim or not in seven steps. The model is based on the reviewer's experience. After this process, it goes through one of two processes. If it is considered as a normal claim, the result of the examination is sent to the NHIS. Then, NHIS reimburses the provider. If the claim is considered to be suspicious, it goes through the manual review. In this process, the reviewers manually examine the claim one by one. Manual review involves two kinds of examination:. the regular examination, and the irregular examination. In the regular examination process, reviewers select several abusive providers and manually review all the claims from them. Here, the abusive providers are selected by the datamining model of HIRA's own device, which uses provider-level information as input variables. In the irregular examination, reviewers select several important and complex claims and review them precisely. If the claim is much more complex than the others, it goes through

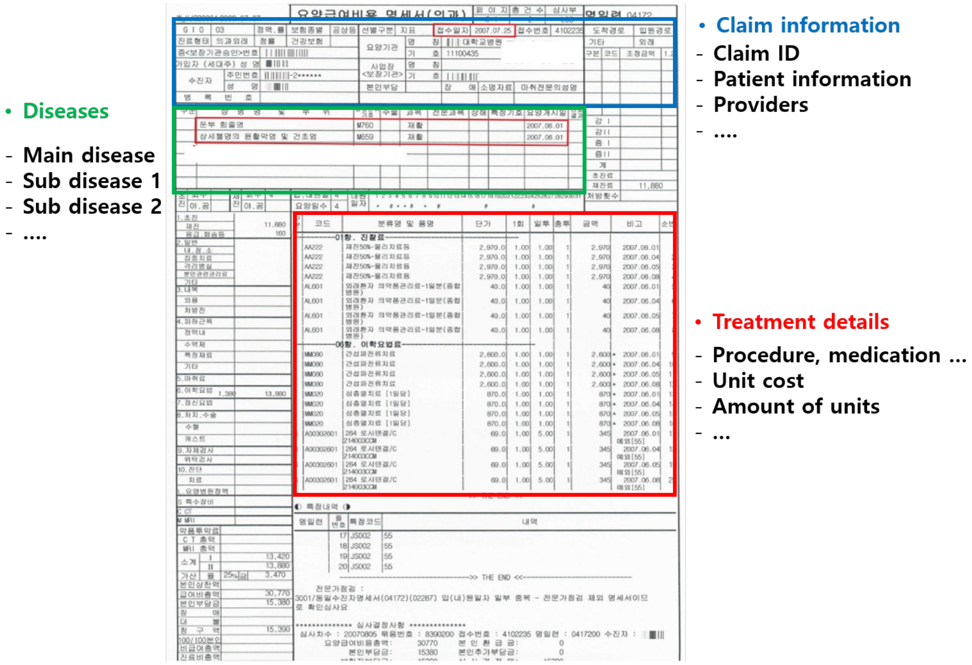
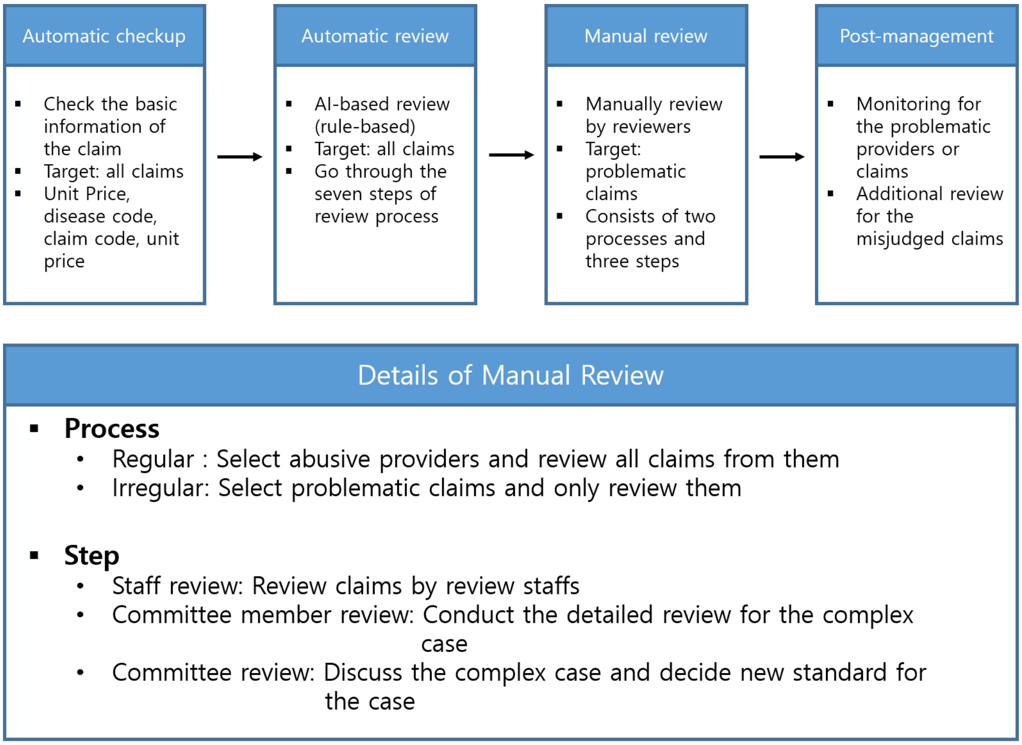

• Claim information
- Claim ID
- Patient information
- Providers
- ....
• Diseases
- Main disease
- Sub disease 1
- Sub disease 2
- ....
• Treatment details
- Procedure, medication ...
- Unit cost
- Amount of units
- ...

Figure 2.5: An example of medical claim of South Korea

the precise examination by the committee members. We illustrate the review process of HIRA in Figure 2.6.

There are several databases that are separately stored within the HIRA data warehouse. Each database stores important information about the insurance claim such as claim information, treatment information, disease information, and review details. The details of the databases that we integrated into a single data are listed in Table 2.2. We did not use all data in the databases. Instead, we extracted records that are relative to the claim filed in 2016. Also, we selected several important variables in consultation with the field experts. Then, we integrated the tables into a single data and preprocessed the resulting table. As a result, we extracted about



Automatic checkup

- Check the basic information of the claim
- Target: all claims
- Unit Price, disease code, claim code, unit price

Automatic review

- AI-based review (rule-based)
- Target: all claims
- Go through the seven steps of review process

Manual review

- Manually review by reviewers
- Target: problematic claims
- Consists of two processes and three steps

Post-management

- Monitoring for the problematic providers or claims
- Additional review for the misjudged claims

Details of Manual Review

- **Process**
  - Regular : Select abusive providers and review all claims from them
  - Irregular: Select problematic claims and only review them
- **Step**
  - Staff review: Review claims by review staffs
  - Committee member review: Conduct the detailed review for the complex case
  - Committee review: Discuss the complex case and decide new standard for the case

Figure 2.6: The review process in HIRA

107 million treatment records.

Then, we selected 18 numerical variables and 18 categorical variables as input variables for modeling. We cannot list all variables that we used, because of the data confidentiality issue. However, we explain three important categorical variables that have high-cardinality: disease codes, special patient code, and the treatment code. In the raw data, the disease codes are in Korean standard classification of diseases (KCD) format. However, we used aggregated codes because there are too many codes to use all of them in modeling. The treatment code variable has also same characteristics. Because of the same reason, we used aggregated codes of treatment codes. Finally, each of them has 1902, 196, 7882 category values in modeling. In spite of this process, these variables still have high cardinality. So, we trained embedding vectors of these variables in modeling, as we presented in subsection 2.3.1.

The class to be predicted is defined as follows. When reviewers label certain treatment to be unnecessary for the patient, the abused amount of the treatment equals the claimed amount for the treatment. When a treatment is considered to be necessary but excessive, the abused amount is less than claimed amount. We define these treatments as abused treatments. Otherwise, we define a treatment with no abused amount as a normal treatment. The information about abused amount is stored in the 'Review details' database in Table 2.2. We report the number of providers, claims, treatments, and class ratio by department in Table 2.3. [112] Due to the data confidentiality issue, we anonymized the name of the department. For modeling and evaluation, we split the data set into train, validation, and test set by 6:2:2 with a similar class ratio. We use the train and the validation set for training the treatment scoring model for each department, and the test set for evaluation.

Table 2.2: Used databases and their details

| Database | Details |
|---|---|
| Claim information | - Basic information of claim<br>ex) claim number, patient information |
| Treatment details | - Treatment or prescription information<br>ex) treatment code, prescription code, daily dosage |
| Review details | - Manual review results<br>ex) review code, abused amount |
| Disease information | - Disease codes related to the claim<br>ex) main disease code, sub disease codes |

Table 2.3: Data statistics

| Department | Number of providers | Number of claims | Number of treatments | Proportion of abuse |
|---|---|---|---|---|
| A | 393 | 820,511 | 58,296,667 | 2.28% |
| B | 255 | 328,406 | 24,122,644 | 4.94% |
| C | 33 | 154,254 | 9,596,701 | 0.89% |
| D | 103 | 165,294 | 6,941,573 | 2.14% |
| E | 156 | 78,740 | 5,016,126 | 1.72% |
| F | 116 | 50,698 | 3,191,256 | 1.97% |

### 2.4.2 Experimental Settings

As we described in subsection 2.3.1, the treatment scoring model is a deep neural network with embedding layers for categorical variables with high cardinality. We create a non-linear decision boundary by activating hidden layers with non-linear activation functions, such as sigmoid, tanh, ReLU [64], ELU [17], LeakyReLU [57]. For categorical variables with high cardinalities, we used different embedding dimension in compliance with their cardinalities. Also, we experimented with various hidden layer size. To prevent overfitting, we used dropout [85] and early stopping techniques [74]. The maximum number of iterations is 200000 and the batch size is 1024. We also experimented with different optimizer such as Adagrad [24], RMSProp [89], and Adam [39]. In all cases, we set the initial learning rate at 0.0002.

Another important issue in this problem is the class imbalance problem. As we can see in Table 2.3, the class ratio is extremely imbalanced. The abuse cases occur rarely. If we do not address this issue properly, the neural network will learn the parameters so that the error is minimized only for the majority class data. Because the loss from the minority class is much less of importance than that of the majority class. In order to prevent this problem, we oversample the minority class data in every mini-batch.

We should also determine how to express categorical variables with high cardinality. The most common approach is to create dummy variables. However, as data grows, it may suffer from the curse of dimensionality [90]. In particular, these variables involves memory issues.

In this paper, we cope with such issues by using two methods. Firstly, we implement the proposed method with Tensorflow package [1]. We also used Compressed Sparse Row methods in Scipy package [91] to convert the dense matrix to sparse one. Then train a logistic regression model. In both cases, we select a model with the largest area under precision-recall curve (AUPRC) in the validation set [21]. We illustrate the process of selecting the best model in Figure 2.7.

### 2.4.3 Evaluation Measure (1): Relative Efficiency

In subsection 2.4.3 and 2.4.4, we elaborate on the devise on the evaluation measure. The baseline model for our experiments is the scoring model employed by HIRA. This model is based on discriminant analysis method with a set of provider-level variables. This model calculates the abuse scores for all providers. Then, reviewers select several suspicious providers based on the scores and examine all claims from them. Otherwise, the proposed method is based on a deep neural network with treatment-level variables. In this subsection, we explain a performance measure named relative efficiency, which quantifies the extent of efficiency improved by using the proposed method over the previous method.

The scatter plots on the left side of Figure 2.8 plot the abuse score against the total abused amount of each provider when evaluated by model A (above) and

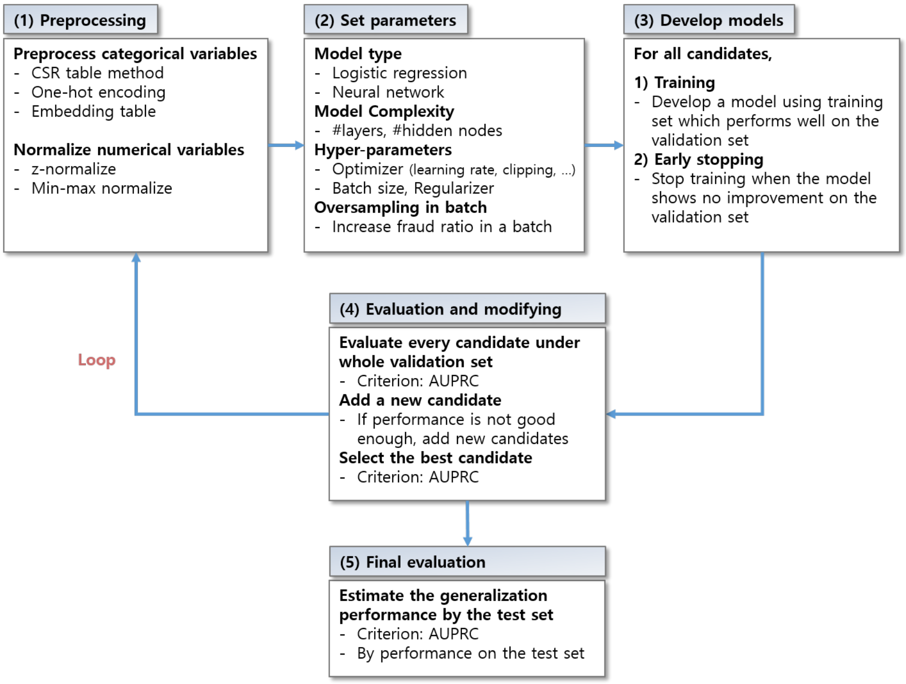

(1) Preprocessing
Preprocess categorical variables
- CSR table method
- One-hot encoding
- Embedding table
Normalize numerical variables
- z-normalize
- Min-max normalize
(2) Set parameters
Model type
- Logistic regression
- Neural network
Model Complexity
- #layers, #hidden nodes
Hyper-parameters
- Optimizer (learning rate, clipping, …)
- Batch size, Regularizer
Oversampling in batch
- Increase fraud ratio in a batch
(3) Develop models
For all candidates,
1) Training
- Develop a model using training set which performs well on the validation set
2) Early stopping
- Stop training when the model shows no improvement on the validation set
(4) Evaluation and modifying
Evaluate every candidate under whole validation set
- Criterion: AUPRC
Add a new candidate
- If performance is not good enough, add new candidates
Select the best candidate
- Criterion: AUPRC
Loop
(5) Final evaluation
Estimate the generalization performance by the test set
- Criterion: AUPRC
- By performance on the test set

Figure 2.7: Training and selecting the best treatment scoring model

model B (below). If the scoring model is well-trained, the model will assign a high score to an abusive provider. That is, in other words, the model score and the actual abused amount should increase in proportion. The scatter plots on the left side of Figure 2.8 shows that the model B is better trained than model $A$. In the meantime, the right panel of Figure 2.8 reports the actual cumulative abused amount in descending order of each model's scores. Suppose that only half of all providers have been examined for abuse cases. According to the right-hand side plot, Model $B$ has detected approximately 80% of the entire pool of abused amount, while Model $A$ has only detected about 50% of abused amount. From this graphical investigation, we can infer that model $B$ examines claims more efficiently than model $A$ does.

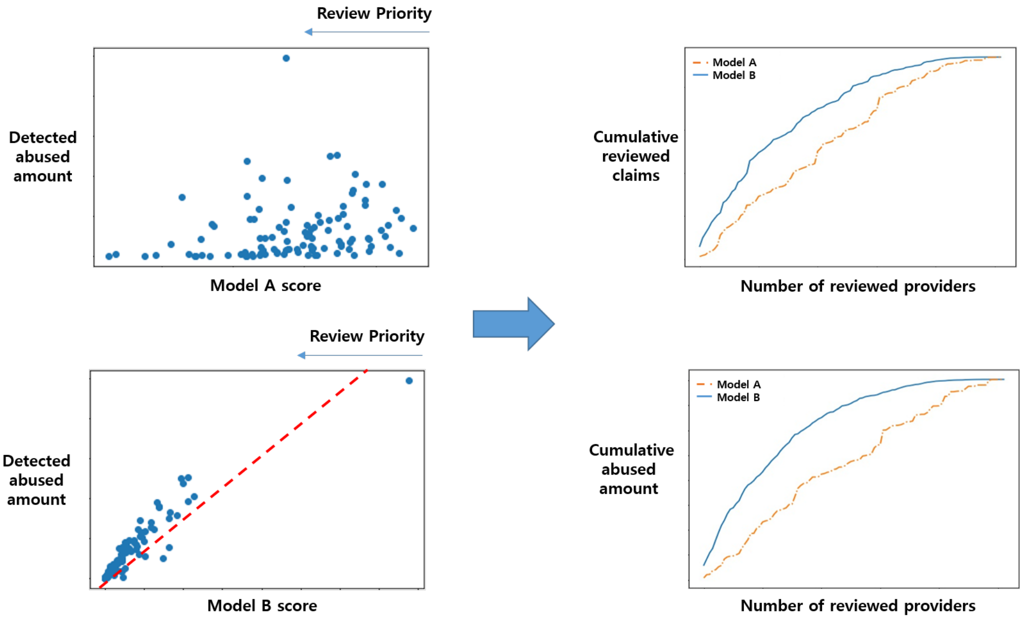


Figure 2.8: The concept of relative efficiency

From now on, we establish the definition of the efficiency of the reviewing process more concretely. First, we define the efficiency of review as the abused amount detected in relation to the efforts required for the examination. Until now, we have regarded the number of examined providers as the efforts. However, this is not enough. Suppose there are two providers, where one submits more claims than the other. In this case, greater efforts are required to review all the claims filed by the former than those by the latter. In other words, the amount of effort to review all the claims varies from provider to provider depending on the number of filed claims. This is the reason why we have to define the efforts as the number of examined claims rather than the number of examined providers. Therefore, in order to quantify the efficiency, we consider both the cumulative number of examined claims and the cumulative abused amount as illustrated in the Figure 2.8.

Mathematically, we express the efficiency of review as follows. Suppose there are $N$ providers for a department, and the number of claims and the number of medical treatments are defined as $m_1, m_2, \ldots, m_N$ and $n_1, n_2, \ldots, n_N$, respectively. Further, suppose that all providers are sorted in the descending order by the abuse score calculated from model A. The number of claims and treatments can then be represented by $m^A_{(1)}, m^A_{(2)}, \ldots, m^A_{(N)}$ and $n^A_{(1)}, n^A_{(2)}, \ldots, n^A_{(N)}$, respectively. In addition, we define $d^A_{(1)}, d^A_{(2)}, \ldots, d^A_{(N)}$ as the abused amount detected for each provider. Then, for providers with the top-$k$ highest scores, the number of claims, treatments and the abused amount are represented by $m_k$,$n_k$ and $d_k$. More specifically, the total number of claims is $M = m_1 + m_2 + \ldots + m_N = m^A_{(1)} + m^A_{(2)} + \ldots + m^A_{(N)}$. Now suppose the reviewers can screen $p\%$ of the total number of claims. That is, in other words, the reviewers can only screen $0.01pM$ claims. Then there exists $h$ that satisfies the following inequalities.

$$\sum_{i=1}^{h} m^A_{(i)} \leq 0.01pM, \quad \sum_{i=1}^{h+1} m^A_{(i)} \geq 0.01pM$$

Here, the detected abused amount is $\sum_{i=1}^{h} d^A_{(i)}$. The efficiency of review is now defined as the total abused amount detected by the reviewer in comparison to the number of reviewed claims. If the reviewers select providers with scores computed by Model A, our proposed model, the efficiency can be represented as follows:

$$e^A_p = \frac{\sum_{i=1}^{h} d^A_{(i)}}{\sum_{i=1}^{h} m^A_{(i)}}$$

We are not at liberty to compute this measure explicitly due to data confidentiality issues. Hence, we replace it with the following term, called relative efficiency, to compare the efficiency of the two scoring models. Mathematically, relative efficiency can be expressed as below:

$$e_p^{A,B} = \frac{e_p^A}{e_p^B}$$

This measure quantifies improvements in efficiency improvement when selecting providers to review with model A as the base for comparison.

The number of providers reviewed may change at every review session. Hence, it is essential to be able to compute efficiency even though the size of providers are varying. We incorporate this idea into the relative efficiency measure and redefine the term as follows.

$$e_p = \frac{e_p^{proposed}}{e_p^{HIRA}}$$

Here, $HIRA$ stands for the previous scoring model that HIRA has been using, and *proposed* stands for the proposed scoring model.

### 2.4.4 Evaluation Measure (2): Precision at $k$

The concept of precision at $k$ refers to the proportion of relevant items in the top-$k$ item set retrieved. It is widely used in information retrieval field to measure the performance. In this study, we re-define the precision at $k$ measure to fit our purpose as follows. Suppose $A_k$ as the set of the institutions with top-$k$% abuse score, and $B_k$ as the set of the providers with top-$k$% abused amount. Let the precision at $k$ represent the proportion of the providers with top-$k$% abused amount in the providers to top-$k\%$ abuse score institution set. Then, mathematically, precision at $k$ can be expressed as below:

$$Pr_k = \frac{|A_k \cap B_k|}{|A_k|}$$

This metric measures the model's ability to detect providers with greater abused amount. In other words, it measures the extent of the model's ability to detect providers with a severely abusive billing pattern.

## 2.5 Results

### 2.5.1 Results in the test set

We illustrate the change of cumulative abused amount at different portions of the reviewed claims at department A in Figure 2.9. Suppose the reviewers can only examine 80% of the total claims. If they select the providers for review based on the score of the previous model, they will select 340 providers to review. In contrast, they will select 220 providers with the proposed model. If they reviewed all claims from 220 providers that the proposed model has recommended, they can detect 1.09 times more abused amount than reviewing all claims from the 340 providers recommended by the previous model. In short, they proposed model is 1.09 times more efficient than the previous model at the 80% level. Similarly, the proposed model is 1.13 times more efficient than the previous model at the 60% level, and 1.26 times more efficient than the previous model at the 40% level. We report relative efficiency values at various levels of proportions of claims reviewed for each department in Table 2.4. In Table 2.4, 'max' means the level when the maximum relative efficiency is achieved. We can see that the relative efficiency is larger or equal to 1 in most cases. It means the proposed model is more efficient than the previous model in most cases.

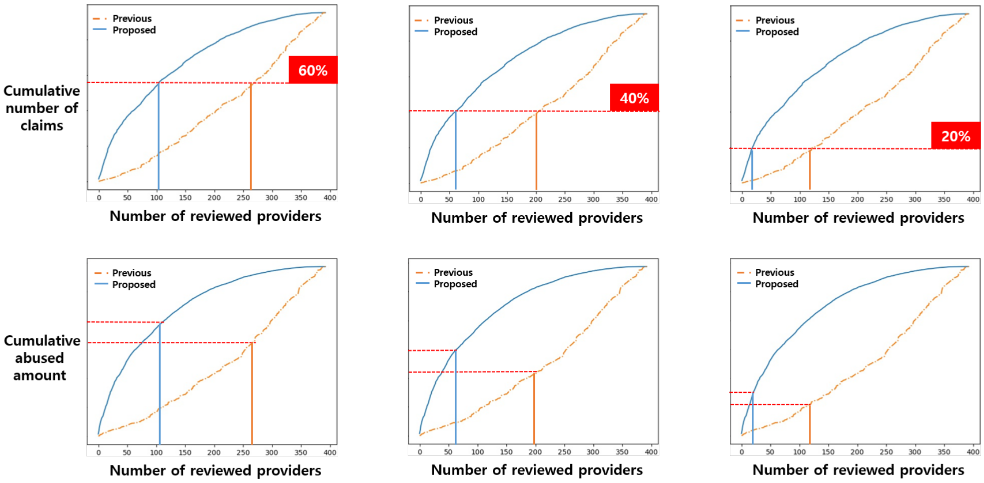


Figure 2.9: Relative efficiency at different levels

One more thing that we can see from Table 2.4 is the tendency that the relative efficiency values tend to grow larger at small $p$ than the larger one. This implies that the proposed model assigns higher scores to the more highly abusive providers, while previous model fails to do so. It is clearly observed in Table 2.5, that at small $k$, the precision at $k$ of the proposed model shows much better performance than that of the previous model.

Table 2.4: Relative efficiency on the test set by department

| Department | $e_{20\%}$ | $e_{40\%}$ | $e_{60\%}$ | $e_{80\%}$ | $e_{MAX}$ |
|---|---|---|---|---|---|
| A | 1.03 | 1.28 | 1.13 | 1.09 | 1.33 |
| B | 3.33 | 1.91 | 1.26 | 1.14 | 3.50 |
| C | 1.95 | 2.10 | 2.10 | 1.19 | 2.10 |
| D | 1.24 | 1.13 | 1.10 | 1.19 | 1.50 |
| E | 1.61 | 1.23 | 1.21 | 1.17 | 1.61 |
| F | 0.87 | 1.18 | 1.09 | 1.23 | 1.76 |

Table 2.5: Precision at $k$ on the test set by department

| Department | $Pr_{10}$ | | $Pr_{20}$ | | $Pr_{30}$ | | $Pr_{40}$ | | $Pr_{50}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* |
| A | 0.00 | **0.70** | 0.05 | **0.82** | 0.16 | **0.84** | 0.24 | **0.92** | 0.37 | **0.90** |
| B | 0.00 | **0.77** | 0.06 | **0.90** | 0.08 | **0.88** | 0.14 | **0.87** | 0.29 | **0.92** |
| C | 0.00 | **0.75** | 0.43 | **1.00** | 0.30 | **1.00** | 0.50 | **0.93** | 0.65 | **0.94** |
| D | 0.09 | **0.73** | 0.43 | **0.71** | 0.42 | **0.90** | 0.57 | **0.91** | 0.64 | **0.90** |
| E | 0.38 | **0.94** | 0.38 | **0.81** | 0.47 | **0.87** | 0.51 | **0.95** | 0.59 | **0.91** |
| F | 0.25 | **0.75** | 0.46 | **0.79** | 0.51 | **0.91** | 0.55 | **0.87** | 0.60 | **0.88** |

### 2.5.2 The Relationship among the Claimed Amount, the Abused Amount and the Abuse Score

The proposed model calculates the abuse score of the provider by summing up the resulting scores from multiplying the output of the treatment scoring model and claimed amount for each treatment. By definition, the abuse score of a provider is variant to the total claimed amount filed by the provider. Without the scores resulting from the scoring model, the abuse score would merely reflect the total claimed amount from the provider. It means that the model only selects the providers with top-$k$ highest claimed amount. However, by including results from the treatment scoring model, the proposed model selects broader types of abuse cases.

This is illustrated in Figure 2.10. It shows the total abused amount and the abuse scores 20 providers with the largest claimed amount. The providers are sorted in descending order by the claimed amount. First, let us look at the relationship between the claimed amount and the abused amount. Providers are sorted in descending order by claimed amount, but the abused amount does not tend to descend. It means that large claimed amount does not mean large abused amount. Likewise, the abuse score does not tend to descend, which means that the score is not simply proportional to the claimed amount. The difference is from the result which is calculated from the treatment scoring model. Due to this term, the bias caused by the claimed amount is reduced. Also, we can see that abuse score moves in accordance with the real abused amount. This means that the abuse score calculated by the proposed model estimate the abuse degree of the provider well.

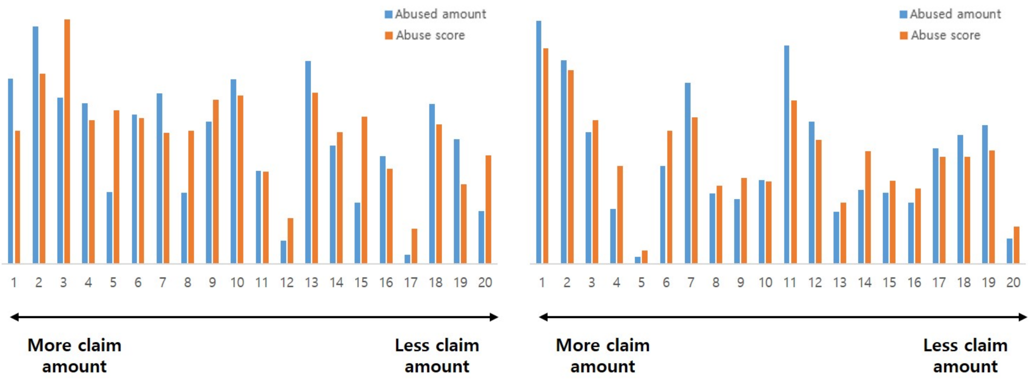


Figure 2.10: The relationship among claimed amount, abused amount and proposed abuse score

### 2.5.3 The Relationship between the Performance of the Treatment Scoring Model and Review Efficiency

In this subsection, we will discuss the performance of the treatment scoring model on the performance of the provider scoring model. In the previous subsection, we built lots of treatment scoring models with various hyper-parameters and select a model that has the best performance in the validation set.

In order to show the impact of the treatment scoring model on the performance of the provider scoring model, we performed the following experiment. First, randomly select a treatment scoring model whose performance is slightly lower than the best one. Then, calculate the performance of the provider scoring model with both the selected one and the best one. We show the impact indirectly by comparing them. We report the performances of both cases in Table 2.6 and Table 2.7. In most cases, the performance of the selected model is slightly less than or similar to the best one. In particular, the relative efficiency is seemed to be very similar. However, remind the relative efficiency is a 'relative' performance measure. There are cases that the difference in relative efficiency by 0.1 means millions of dollars. Therefore, it is not a small difference.

Table 2.6: Relative efficiency of the randomly chosen model and the best model

| Department | $e_{20\%}$ | | $e_{40\%}$ | | $e_{60\%}$ | | $e_{80\%}$ | | $e_{MAX}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* |
| A | **1.05** | 1.03 | 1.22 | **1.28** | 1.10 | **1.13** | 1.06 | **1.09** | 1.25 | **1.33** |
| B | **3.07** | 1.33 | 1.84 | **1.91** | 1.25 | **1.26** | 1.13 | **1.14** | 3.43 | **3.50** |
| C | 1.95 | **1.95** | 2.10 | **2.10** | 2.10 | **2.10** | 1.19 | **1.19** | 2.10 | **2.10** |
| D | 0.98 | **1.24** | 0.89 | **1.13** | 1.03 | **1.10** | 1.04 | **1.19** | 1.33 | **1.50** |
| E | 1.52 | **1.61** | 1.22 | **1.23** | 1.20 | **1.21** | 1.16 | **1.17** | **1.56** | 1.61 |
| F | 0.64 | **0.87** | 0.98 | **1.18** | 1.09 | **1.09** | 1.22 | **1.23** | 1.22 | **1.76** |

### 2.5.4 Treatment Scoring Model Results

In this subsection, we will discuss how the structure of the treatment model affects the performance of the proposed model. In the previous subsections, we compared the proposed method to previous method that is based on the provider-level variables.

Table 2.7: Precision at $k$ of the randomly chosen model and the best model

| Department | $Pr_{10}$ | | $Pr_{20}$ | | $Pr_{30}$ | | $Pr_{40}$ | | $Pr_{50}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* |
| A | 0.68 | **0.70** | 0.79 | **0.82** | 0.81 | **0.84** | 0.87 | **0.92** | 0.87 | **0.90** |
| B | **0.81** | 0.77 | 0.84 | **0.90** | 0.86 | **0.88** | 0.84 | **0.87** | 0.90 | **0.92** |
| C | 0.75 | **0.75** | 0.86 | **1.00** | 0.90 | **1.00** | 0.93 | **0.93** | 0.94 | **0.94** |
| D | 0.64 | **0.73** | 0.62 | **0.71** | 0.81 | **0.90** | 0.81 | **0.91** | 0.87 | **0.90** |
| E | 0.88 | **0.94** | 0.75 | **0.81** | **0.89** | 0.87 | 0.92 | **0.95** | 0.91 | **0.91** |
| F | 0.75 | **0.75** | **0.83** | 0.79 | 0.89 | **0.91** | 0.83 | **0.87** | 0.86 | **0.88** |

However, since we exploit the treatment-level variables in the proposed method, it is not appropriate to compare directly between two models. Instead, we compared the proposed model to logistic model that uses treatment-level variables. By this comparison, the proposed structure is appropriate for the treatment scoring model.

In order to handle the categorical variables with high cardinality, we applied CSR method provided by the Scipy package. Also, we experimented with various class weight in training logistic models. Then, we selected a model that shows the best performance in the validation set.

In Table 2.8, we reported the AUPRCs from the logistic regression and the proposed treatment scoring model for each subject in the test set. As we can see, the proposed model performs much better than the logistic regression model in every case. For the case of logistic regression, categorical variables with high cardinality are one-hot encoded. So, the dimension of the data becomes much larger than before. As a result, it requires much more complex computation while the performance does not meet with the complexity of the model. However, the proposed model learns not only the network parameters but also the embedding function to minimize the error. This might be a reason that led to much better performance as compared to the logistic regression model.

In subsection 2.5.3, we have shown that the performance of the treatment scoring model affects the performance of the provider scoring model. From this result, both model complexity and learning the embedding function of categorical variables with high cardinality play an important role in determining the performance of the model.

Table 2.8: The AUPRC of the best treatment scoring model

| Department | Logistic regression | Proposed model |
|---|---|---|
| A | 0.24 | **0.60** |
| B | 0.41 | **0.72** |
| C | 0.44 | **0.73** |
| D | 0.31 | **0.63** |
| E | 0.30 | **0.69** |
| F | 0.25 | **0.63** |

### 2.5.5 Post-deployment Performance

Suppose a situation that reviewers select abusive providers from claim data in previous year and examine all claims from them in this year. If the scoring model performs well and the data distribution is similar between two years, the reviewing process may be efficient. If then, the proposed model can be used in reality. So, we experimented with the claim data filed in 2016 and 2017. We trained models and selected abusive providers with the claim data in 2016. Then, we evaluated the performance

Table 2.9: Data statistics used for evaluating post-deployment performance

| Department | Number of institutions | Number of claims | Number of treatments | Proportion of abuse |
|---|---|---|---|---|
| A | 259 | 280,083 | 24,274,388 | 0.61% |
| B | 101 | 77,247 | 5,875,745 | 0.64% |
| C | 24 | 14,706 | 1,029,135 | 0.63% |
| D | 76 | 75,922 | 3,646,965 | 0.52% |
| E | 128 | 31,214 | 2,121,534 | 0.49% |
| F | 90 | 28,902 | 1,915,349 | 0.67% |

with the data in 2017. There may some providers that exists in 2017 but not in 2016 and vice versa. We excluded such providers in the experiment. Table 2.9 lists the summary statistics that we used. We report the performance in Table 2.10 and Table 2.11. As we can see from these tables, the proposed model is more efficient than the previous model. Also, it detects abusive providers well.

Table 2.10: Relative efficiency in 2017 based on the results of 2016

| Department | $e_{20\%}$ | $e_{40\%}$ | $e_{60\%}$ | $e_{80\%}$ | $e_{MAX}$ |
|---|---|---|---|---|---|
| A | 1.37 | 1.32 | 1.12 | 1.09 | 1.38 |
| B | 5.95 | 2.10 | 1.43 | 1.11 | 6.10 |
| C | 1.00 | 3.45 | 2.10 | 1.70 | 3.53 |
| D | 1.60 | 1.14 | 1.21 | 1.13 | 1.80 |
| E | 1.41 | 1.26 | 1.22 | 1.22 | 1.43 |
| F | 0.82 | 0.99 | 0.98 | 1.15 | 1.39 |

Table 2.11: Precision at $k$ in 2017 based on the results of 2016

| Department | $Pr_{10}$ | | $Pr_{20}$ | | $Pr_{30}$ | | $Pr_{40}$ | | $Pr_{50}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* |
| A | 0.00 | **0.65** | 0.06 | **0.71** | 0.14 | **0.72** | 0.24 | **0.78** | 0.37 | **0.81** |
| B | 0.00 | **1.00** | 0.00 | **0.76** | 0.00 | **0.81** | 0.07 | **0.88** | 0.24 | **0.82** |
| C | 0.33 | **0.33** | 0.40 | **0.60** | 0.38 | **0.88** | 0.50 | **0.80** | 0.58 | **0.92** |
| D | 0.00 | **0.50** | 0.31 | **0.69** | 0.39 | **0.65** | 0.45 | **0.84** | 0.63 | **0.84** |
| E | 0.39 | **0.69** | 0.35 | **0.77** | 0.46 | **0.74** | 0.48 | **0.71** | 0.58 | **0.80** |
| F | 0.11 | **0.56** | 0.39 | **0.78** | 0.57 | **0.82** | 0.59 | **0.81** | 0.58 | **0.78** |

## 2.6 Summary

Healthcare insurance companies manually review all the medical claims to detect abuse in order to avoid issuing unnecessary compensations. However, as the number of claim filings grow exponentially, the cost of manual review increases astronomically, which calls for a more efficient review process. By efficiency, we set our objectives to detect as much abused amount correctly as possible with minimum effort. It is particularly important to effectively screen out abusive medical providers, as they are more likely to prescribe unnecessary treatments to the patients. Such a screening process, in turn, will require a scoring scheme which that measures the degree of abuse.

In this chapter, we propose the very first model that scores abusive billing patterns of providers using the medical treatment data. The proposed model consists of two steps: (1) training a neural network to compute the likelihood of abuse for each treatment, and (2): calculating the abuse score for each treatment and aggregating the results up to the provider level. The abuse score for each treatment is calculated by multiplying the neural network result with claimed amount. Experiment results show that our proposed model scores abusiveness better than the model with features summarized at the provider-level.

The main contribution of this chapter lies in that it is one of the first research detecting the abusive provider using medical treatment data, which is the finest-grained level data in terms of medical claims. Previous studies extract the provider-level variables such as the number of prescriptions per day or the average cost per claim and use these variables for training. This way, the model cannot properly account for information apparent only at the claim or treatment-level. In contrast, we fully exploit the fine granularity of the treatment data to train the model. The experiment results show that the proposed model performs better than the model with provider-level variables. In addition, we devise performance metrics, relative efficiency and precision at k, to quantify the efficiency improvement. Using these metrics, we show that the reviewers can review more efficiently by looking at providers determined to be suspicious of abuse by the proposed model as compared to examining those selected by the model with provider-level variables. Finally, we show that the performance of the treatment scoring scheme is important to computing an effective abuse score. This implies that training better neural network results in better performance.

If a provider is chosen to be abusive and all its claims are reviewed, it will not be reimbursed for the amount determined to be abused, which will result in the loss of the provider. Consequently, the provider does not want to be selected, which in turn reduces the waste of health insurance, so that insurance companies can reduce unnecessary costs. However, as the medical environment continues to change, it also creates forms of abuse that did not exist before. Previous scoring methods using the existing provider-level variables cannot adapt to this changing pattern of abuse. On the contrary, the proposed model scores abusivesness while adapting to changes in abuse patterns through regular retraining.

There are two potential limitations to our model. Firstly, we assume that the filed claims are uniformly distributed across time. Our experiment splits the entire data set into the training, validation, and test sets, of which the underlying assumption is that learning is not contingent upon time. However, from the practical point of view, such an assumption may not hold true for some cases. In the next chapter, we address this issue in greater detail and propose a model which accounts for seasonality.

Another limitation of the current model is that it does not explicitly consider the association relationship between diseases and treatments, one of the most significant factors in reviewing claims. In chapter 4, we discuss this issue in detail and propose a model explicitly dealing with this relationship.

# Chapter 3

# Detection of overtreatment by Diagnosis-related Group with Neural Network

## 3.1 Background

In chapter 2, we introduce the very first method, to our best knowledge, to detect abusive providers by using medical treatment data, which is the lowest level of healthcare information available. We show the review process would be more efficient if the field reviewers give priority to the candidates of abusive providers selected by the proposed model instead of those screened by the previous method. The proposed model computes the degree of the provider's abusiveness numerically, which helps interpret the detection result. The key assumption underlying our model is that the distribution of claim data is similar between the training set and the test set.

Before we discuss this issue, let us define the distribution of the claim data. We believe that the most important information from the claim filings is diseases of diagnose and the prescribed treatments. In the perspective of our data, then we consider a claim as a single value from a distribution of claims. Under this setting, we assert that the representative value of a claim should include both disease and treatment information. Here, every disease and treatment information does not have to be included. We can represent the claim by main disease and several important treatments only. Given these three requirements, we believe that the diagnosis-related group (DRG) is an appropriate measure to serve as the aforementioned representative value. It includes information about the main disease the patient is diagnosed of, as well as the treatments the practitioner has prescribed.

The previous model used by HIRA assumes that claims data follows a homogeneous distribution, which is too strict of an assumption to assert to hold true in reality. One well-known counterexample is seasonality. A handful of diseases, flu, for example, show period surge of infected patients for a specific period of time, a characteristic to which we refer as seasonality. Seasonality implies that the distributions of claims may shift by time, according to its seasonal surge and dissolution. In fact, as we can see by observing Figure 3.1 that this assumption is realistic.

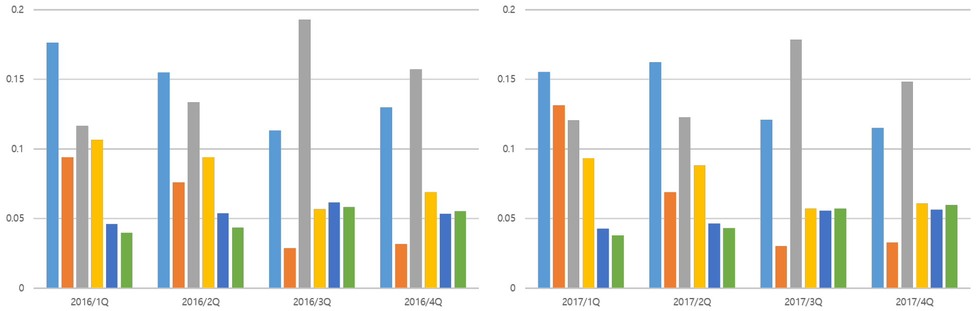


Figure 3.1: The distribution of the patient group in a department

Suppose we train the model ignoring these seasonal patterns. The model's primary objective to minimize the training error; consequently, the model will focus solely on claim data whose DRG-code is the majority. Hence, the model won't perform well if it is trained to learn to compute the degree of abuse under the homogeneity assumption. A candidate solution is to train the model using observations from the same point of time every year. For example, one may choose to use claims filed in during the 1st financial quarter of last year for training and then use claims the 1st quarter of this year to estimate the score for the degree of abuse. Such an approach is still not safe from events that randomly or unexpectedly taking place, such as medical inventions and innovations. Suppose an ingenious clinical innovation has intervened between the training period and the scoring period. Then, information used for training is outdated and there are new forces governing the features of claims filed in this year, leading the model to poor performance.

In order to account for seasonality during learning, we train our treatment classification model with claims data grouped by DRG code. The DRG system is a type of patient classification scheme (PCS) which provides a means of relating the type of patients a hospital treats to the costs incurred by the hospital [11]. DRG system categorizes patient episodes by controlling the fundamental variations, which are assumed to be always present, among patients. Claims with the same DRG code include similar disease or treatment.

If we train models by DRG code, then the distribution of claims will be more homogeneous as compared to the existing model. Even if distinct seasonal characteristics, peculiar to each disease, exist, because the model is trained by similar diseases. Consequently, our model will produce results robust to seasonality. Suppose we have to detect abuse in medical treatments in department $A$. Also, suppose every claim in the department has one of two DRG codes: $D_a, D_b$. In the training set, the number of claim data with the DRG code $D_a$ is much larger than that of $D_b$. In this case, the model will be trained to minimize the training error from the data with the DRG code $D_a$. It means the training error from the data with the

DRG code $D_b$ is ignored relative to that with the DRG code $D_a$. However, suppose the number of claim data with DRG code $D_b$ is much larger than that of $D_a$ in the test data. In this case, the trained model cannot classify the data with DRG code $D_b$ and the performance of the model will decrease. However, if models are trained separately for DRG code $D_a$ and $D_b$, the performance will not degrade since data distribution in each data set is similar between the training set and the test set.

In this chapter, we propose to run the treatment classification model by DRG code unit. If then, the model can classify the treatment robust to seasonality. The DRG system has been used in patient classification. It has also been serving as the unit of the DRG-based payment system and as the standard of comparing medical institutions. Our work show the possibility that the DRG system can be used in the review process in healthcare insurance.

The rest of the chapter is organized as follows. In section 3.2, we introduce seasonality in disease and the concept of the DRG system. Section 3.3 provides detailed descriptions of the proposed model. In addition, we introduce strategies to compare performance between our model and the method that is suggested by Lee et al. [47]. In section 3.4, we elaborate on experiment settings. We also provide detailed description of the data and the preprocessing steps in this section. Section 3.5 reports experiment results. Finally, section 3.6 concludes the paper.

## 3.2 Literature review

### 3.2.1 Seasonality in disease

In public health, seasonality is a feature characterized by the surge of a certain disease recurring at a particular time period ([28], [58]). A variety of infectious diseases, such as influenza, as well as some respiratory diseases which are non-infectious, exhibit seasonality.

Even though the awareness for seasonality has existed for a while in the research field, the underlying mechanism of seasonality has not been fully explained. Clear understanding of seasonality will certainly prove beneficial for public health in many different aspects. Fisman [28] claimed that there are four major benefits that may rise from understanding the full mechanism of seasonality: *(1) improved understanding of host and pathogen biology and ecology, (2) enhanced accuracy of surveillance systems, (3) improved ability to predict epidemics and pandemics, (4)better understanding of the long-term implications of global climate change for infectious disease control.* To shed more realistic light on the potential benefits, we take the example of the two viral respiratory illnesses: severe acute respiratory syndrome (SARS) and coronavirus disease 19 (COVID-19). These two diseases are quite similar in a sense that their main agent of contagion is the coronavirus, which is a type of an enveloped RNA virus. If seasonal features associated with the spread of SARS were fully characterized, the results of which may serve as the basis to infer/predict the seasonality of COVID-19. Then, resources may have been allocated accordingly to detect and prevent the disease in a timely manner.

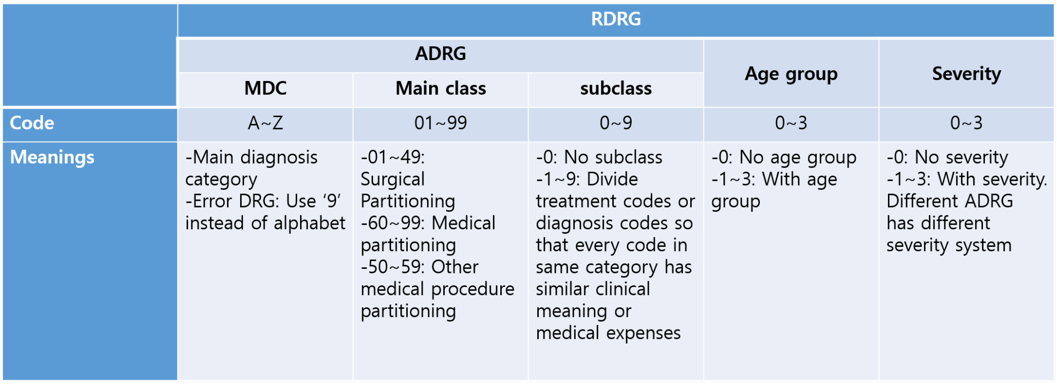
### 3.2.2 Diagnosis related group

Diagnosis-related group (DRG) is one brank of the patient classification system (PCS), which classifies patients in perspective of clinical records and medical resource consumption patterns such as diagnosis, procedures, or functional status [11]. It was first devised in Yale University in the late 1960s. Originally, the objective of

DRG was to create an efficient method for monitoring the quality of patient care and the utilization of service for each hospital. Additional adjustments were continuously made to the system since its first invention, raising its quality to the current level. Now, DRG is exploited in various ways, including hospital-to-hospital comparisons, patient classification, and evaluation of medical institutions. At the same time, it is also used as a unit of the bundled-payment system for healthcare insurance. Bundled-payment system is known to compensate for the shortcomings of the fee-for-service payment system which has been popular of choice. Under the fee-for-service payment system, the insurance company must reimburse for all the treatments provided to the patient. It is more likely to lead to over-treatment, since the provider can enjoy greater reimbursement by performing additional procedures. In contrast, under the bundled-payment system, each patient is classified by the DRG code, and the insurance company only has to compensate for the amount predefined for the subject patient category. In other words, regardless of the number of treatments performed by the provider, the insurance company compensates only for the pre-determined, fixed amount. This, by design, deters providers from over-treatment.

The Korean diagnosis-related group (KDRG) is a modified version of DRG, adjusted to reflect the peculiarity in the medical practice in Korea [38]. The first version of KDRG, KDRG v1.0, was first devised in 1986. Now it is updated to KDRG v4.3 with 2,753 codes for classifying the patients. These codes are constructed by combining Korean classification of diseases (KCD) and treatment codes.

The formation of the DRG code begins by splitting up all the principal diagnoses available into 23 main diagnostic categories (MDC). Then, the MDCs are subdivided either into medical or to surgical categories. For example, a patient is classified as surgical if the prescription on his/her claim includes surgery/operations. Otherwise, the patient is classified as a medical case. Surgical cases are further divided into the groups of finer granularity based on the precise surgical approaches performed; medical cases, based on the exact principal diagnosis. The DRG code assigned by this process is called as the Adjacent DRG (ADRG). In order to classify patients as accurately and appropriately as possible, the age group, as well as the complication and comorbidity factors, are also considered as the classification criteria. The final DRG code resulting after the whole process is called the Refined DRG (RDRG) code. The summary of the KDRG structure is shown in Figure 3.2.



| | RDRG | | | | |
|---|---|---|---|---|---|
| | ADRG | | | Age group | Severity |
| | MDC | Main class | subclass | | |
| Code | A~Z | 01~99 | 0~9 | 0~3 | 0~3 |
| Meanings | -Main diagnosis category<br>-Error DRG: Use '9' instead of alphabet | -01~49: Surgical Partitioning<br>-60~99: Medical partitioning<br>-50~59: Other medical procedure partitioning | -0: No subclass<br>-1~9: Divide treatment codes or diagnosis codes so that every code in same category has similar clinical meaning or medical expenses | -0: No age group<br>-1~3: With age group | -0: No severity<br>-1~3: With severity. Different ADRG has different severity system |

Figure 3.2: The structure of KDRG

## 3.3 Proposed method

This section details the structure of the proposed model which leans to classify the entered treatment to be normal or not. In this study, we use the treatment data, grouped by DRG code, for training as well as for inference. This approach distinguishes our proposed model from the treatment scoring model, as found in Lee et al. [47], which grouped treatment data by the practice department.

### 3.3.1 Training a deep neural network model for treatment classification

The proposed model employs a neural network structure through which a given treatment classified to be normal or to be abused. The input data for the model is heterogeneous, containing both numerical and categorical information. Numerical variables include the unit price of the treatment or the amount of dosage per day, while gender, age group, or the associated treatment codes are the examples of the categorical variables. In order to make the best use of such data as a valid input to a neural network, categorical information must be represented in a form of a numerical vector. One of the most common approaches is to one-hot encode by the given categories. It is not, however, an appropriate approach in this case, because there exist some categorical variables that are of a high-cardinality. If these variables are one-hot encoded, the dimension of the data would explode, hence the suffer from the curse of dimensionality. In our model, we rely on an embedding function, instead, to represent these heterogeneous variables as a vector. Our proposed model trains the embedding function during the training phase. Classification error is back-propagated to the embedding layers as well as the hidden layers.

We describe our model mathematically as follows. Define a medical treatment $t^{'} = [n_1, n_2, \ldots, n_k, c_1, c_2, \ldots, c_l]$, where $n_i$ represents the value of the numerical variable $v_i$ and $c_j$ represents the value of categorical variable $v_j$. We define $\boldsymbol{d}_j$ as the one-hot encoding vector of $c_j$. We represent a categorical variable $v_j$ with value $c_j$ as $\boldsymbol{x}_j = \boldsymbol{d}_j^T$. Otherwise, we compute an embedding vector for the corresponding categorical variable of high-cardinality, represented by $\boldsymbol{x}_j = \boldsymbol{d}_j^T \boldsymbol{W}_j$. Here, $\boldsymbol{W}_j$ stands for the embedding function of the categorical variable $v_j$.

Another important candidate of the heterogeneous input variables, which calls for extra-care, is the multi-valued categorical variable. A multi-valued categorical variable is defined as a variable that has more than one values for each entity. In our study, the major case of the multi-valued categorical variable may be found where a patient is diagnosed to carry more than one diseases. Such cases may be discovered by looking up cases where a practitioner prescribes treatments that may be associated with all the diseases the patient may be suffering from. Since there does not exist the grounds for the identification of the casual relationship between the rich variety of symptoms and the diseases causing these symptoms, as well as the effect of the prescription of the treatments to the corresponding symptoms, it is highly likely to prescribe and practice treatments in response to as many as the candidates of the diseases the subject patient may carry. Such a practice give rise to the multi-valued categorical variables in the input data. In order to effectively represent these variables as numerical vectors, we first embed them through our embedding model and then average the resulting embedding vectors by disease category

We express aforementioned process mathematically as follows. Suppose a given categorical variable $v_j$ is a multi-valued categorical variable. That is, a medical treatment variable $t^{'}$ is represented by $[n_1, n_2, \ldots, n_k, c_1, c_2, \ldots, [c_{j1}, c_{j2}, \ldots, c_{jm_t}], \ldots, c_l]$, where $[c_{j1}, c_{j2}, \ldots, c_{jm_t}]$ is the value of the multi-valued categorical variable $v_j$ for the corresponding treatment $t$. Here, $m_t$ represents the number of values in the variable. It is different by each treatment. Then, we compute the embedding vector for $k$-th value in multi-valued categorical variable and denote it as $\boldsymbol{x}_{jk} = \boldsymbol{d}_{jk}^T \boldsymbol{W}_j$.

Finally, the embedding vector is averaged by the disease category, which is denoted as follows.

$$\boldsymbol{x}_j = \frac{1}{m_t}\sum_k \boldsymbol{x}_{jk} = \frac{1}{m_t}\sum_k \boldsymbol{d}_{jk}^T \boldsymbol{W}_j$$

In summary, the embedding vector of a given heterogeneous categorical variable is defined as follows:

$$\boldsymbol{x}_t = \begin{cases} \boldsymbol{d}_j^T \boldsymbol{W}_j & \text{single-valued, high-cardinality} \\ \frac{1}{m_t}\sum_k \boldsymbol{d}_{jk}^T \boldsymbol{W}_j & \text{multi-valued, high-cardinality} \\ \boldsymbol{d}_j^T & \text{single-valued, low-cardinality} \end{cases} \tag{3.1}$$

Given above representation, we now define the input data for the neural network as following:

$$\boldsymbol{t} = [n_1, n_2, \ldots, n_k, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_l]$$

The output of the neural network $\hat{y} = f_{model}(\boldsymbol{t})$ is calculated by back-propagating the training loss to both the hidden layer and the embedding layer. Then, the embedding function as well as the parameters for the entire network are updated accordingly.

### 3.3.2 Comparing the Performance of DRG-based Model against the department-based Model

The difference between the treatment scoring model as proposed by Lee et al.[47] and our method roots from data used for training and inference. Lee et al.[47] suggests grouping data by department for training and inference. In contrast, we group the

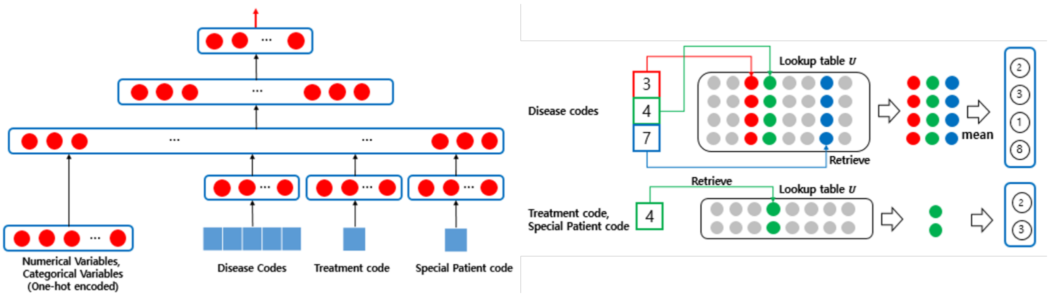


Figure 3.3: The structure of treatment classification model

input data by DRG codes in attempts to reflect the homogeneity of data, while retaining the robustness to seasonality. In this section, we suggest ways to compare out DRG-based model against the department-based model.

Denote the medical treatment set for a given department A for training as $X_A^{trn} = \{\boldsymbol{t}_{A1}, \boldsymbol{t}_{A2}, \ldots, \boldsymbol{t}_{An_A}\}$, and the trained model from $X_A^{trn}$ is represented as $f_A$. Moreover, let the medical treatment set with DRG code $k$, for training, be denoted $X_k^{trn} = \{\boldsymbol{t}_{k1}, \boldsymbol{t}_{k2}, \ldots, \boldsymbol{t}_{kn_k}\}$, and the trained model from $X_k^{trn}$, as $f_k$. The treatment set for department A for inference is represented as $X_A^{inf} = \{\boldsymbol{t}^{'}_{A1}, \boldsymbol{t}^{'}_{A2}, \ldots, \boldsymbol{t}^{'}_{Am_A}\}$, while treatments in the inference set with a DRG code $k$ denoted as $X_k^{inf} = \{\boldsymbol{t}^{'}_{k1}, \boldsymbol{t}^{'}_{k2}, \ldots, \boldsymbol{t}^{'}_{km_k}\}$. Now, suppose there appears DRG codes $a, b, \ldots, k$ in the given department A. Then, we can train $f_a, f_b, \ldots, f_k$ by exploiting $X_a^{trn}, X_b^{trn}, \ldots, X_k^{trn}$. Now, for treatments that are prescribed in the department A with DRG code $i$, we represent them, with $X_{Ai}^{inf}$. It can easily be seen that the treatment set prescribed by department A is the union of $X_{Ai}^{inf}$. Mathematically,

$$X_A^{inf} = \bigcup_i X_{Ai}^{inf} = \bigcup_i \{\boldsymbol{t} | \boldsymbol{t} \in X_A^{inf}, \boldsymbol{t} \in X_i^{inf}\}$$

We denote the classification result of $X_A^{inf}$ through the model $f_A$ is denoted as $\hat{Y}_{DEP} = f_A(X_A^{inf}) = \{f_A(\boldsymbol{t}) | \boldsymbol{t} \in X_A^{inf}\}$. At the same time, the classification result of $X_{Ai}^{inf}$ via the model $f_i$ is $\hat{Y}_{Ai} = f_i(X_{Ai}^{inf}) = \{f_i(\boldsymbol{t}) | \boldsymbol{t} \in X_{Ai}^{inf}\}$. We concatenate $\hat{Y}_{Aa}, \hat{Y}_{Ab}, \ldots, \hat{Y}_{Ak}$ and denote the resulting representation as $\hat{Y}_{DRG}$. Finally, we compare $\hat{Y}_{DEP}, \hat{Y}_{DRG}$, against the true label in order to evaluate the two models' performance. From now on, we define department-based model as the model for calculating $\hat{Y}_{DEP}$ and refer to it as the DEP model. Similarly, we define the DRG-based model as the model for calculating $\hat{Y}_{DRG}$ and refer to it as the DRG model. Figure 3.4 illustrates the entire process.

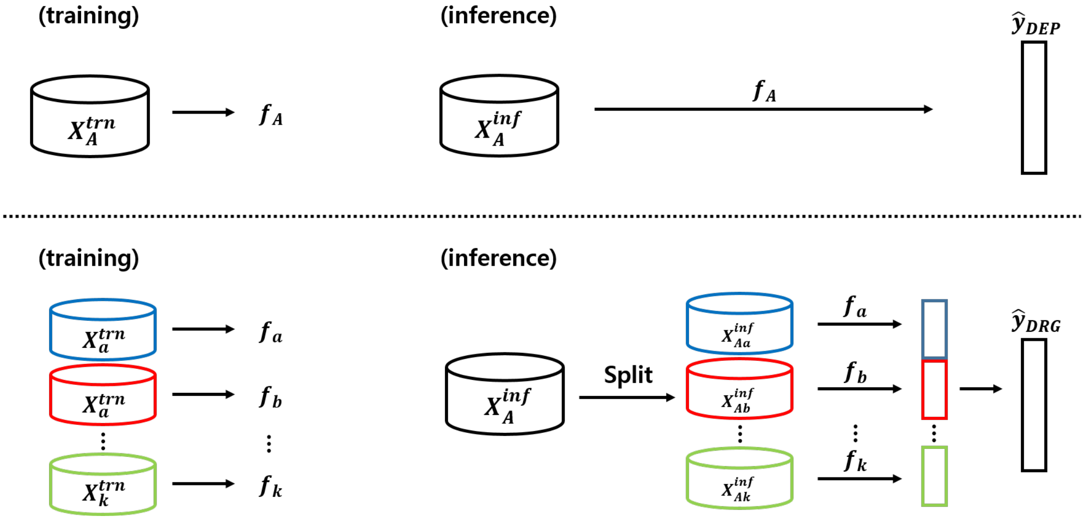


Figure 3.4: Comparison between DEP model and DRG model

## 3.4 Experiments

We evaluate our model on real data which were submitted to HIRA in 2017. We report the performance of our model, with Lee et al.'s [47] as the baseline. The previous method is a scoring model trained with data grouped by department. In subsection 3.4.1, we provide a detailed description of the data as well as the preprocessing steps. We elaborate on performance evaluation metrics in 3.4.2. Finally, training details are presented in 3.4.3.

### 3.4.1 Data Description and Preprocessing

As described in subsection 2.4.1, we worked with several databases that were separately stored within the HIRA data warehouse. Each database stores important features about insurance claims such as claim information, diseases diagnosed and treatments assigned, and the review entailments by the agency. We list the detailed description of each database is presented in Table 2.2. We extracted claim records filed in 2017 that were manually reviewed. [111] During the process, we consulted the on-site field experts and included variables advised as significant by them.

Table 3.1: Data statistics

| Department | Number of Representative DRG codes | Number of claims | Number of treatments | Overtreatment ratio |
|---|---|---|---|---|
| A | 15 | 316,761 | 22,410,573 | 2.13% |
| B | 6 | 45,000 | 2,898,893 | 2.00% |
| C | 2 | 296,238 | 23,331,836 | 5.58% |
| D | 3 | 113,587 | 6,922,504 | 0.67% |
| E | 4 | 169,374 | 7,550,371 | 1.61% |

Following upon the compilation of data, we grouped the resulting claim records by department, as well as the DRG 3-digit code. The reason why we use DRG 3-digit code, instead of the RDRG which is of 6-digit, is because RDRG codes are of too fine granularity. If data is grouped by RDRG codes, then a handful of classes will be empty since grouping is too specific, hence insufficient for learning. As a result, we resorted to DRG 3-digit codes for grouping.

The final complication towards which we should carefully approach during analysis is that more than one DRG 3-digit code may appear even after we group claims by department. For example, given our dataset, approximately two hundred 3-digit DRG codes are observed from the claims filed for the department of internal medicine in 2017. However, the kick here is that the number of treatments for each DRG code follows a long-tail distribution. In other words, a large number of claims cases are associated with only a handful of “important DRG codes”, while few claim reports occur for most of the rest of the DRG codes that are “relatively less important”. Ideally, one would like to models for all the 3-digit DRG codes uniformly across the training phase and compare the performance as illustrated in section 3.3. However, it is impossible, since there are not enough treatment cases with 3-digit DRG codes observed to train the models. Given the restriction on the observed 3-digit DRG codes, we select DRG codes that make up the majority in each department and train the model. Then, we make DRG models using the data that corresponds to the DRG codes. Also, make DEP models using the data corresponds to the data filed from the department and having such DRG codes. Then, we compare the DEP models and DRG models as we already presented in subsection 3.3.2.

The resulting data is processed further following the two important preprocessing schemes: grouping treatments, and integrating treatment codes. First, we cate-

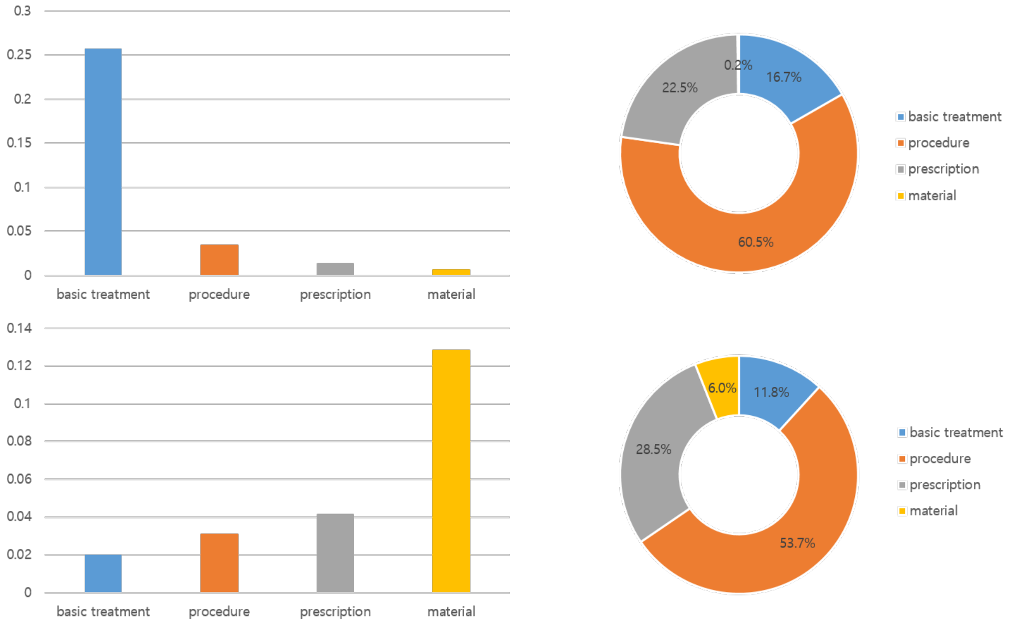
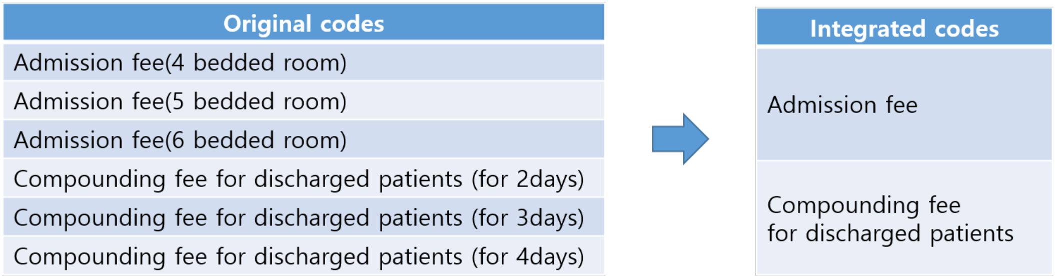


Figure 3.5: The abuse ratio and the distribution of the treatment of two DRG codes

gorize the treatments into four separate groups. The logic for such a process is as follows: suppose that there exists a patient who has received a spine surgery. We assume that the treatments prescribed and practiced for the patient may be compartmented into four distinct categories: the basic treatment, medical procedure, the prescription, and the recovery materials. The basic treatment category includes simple, potentially recurrent medical practices such as admission, consultation, nursing, or providing meals. The medical procedure group entails what practitioners actually conducted on a patient, such as X-rays, MRI examinations, or an operation. The prescription category groups the details of drugs prescribed by the practitioners, such as, for example, the nonsteroidals anti-inflammatory drug. Finally, the recovery materials categorizes all materials needed to recover. A major example of the recovery materials include those for orthosis.

The distribution of DRG codes in data, as well as the class groups, after categorization as described above is quite unbalanced. We take an example and illustrate such a case of class imbalance in Figure 3.5. Figure 3.5 illustrates the class ratio and the distribution of each treatment group of the data with the DRG code. The upper figures are the ratio and the distribution of the data with DRG code of A, while the lower figures are those of the data with DRG code of B. Here, DRG code A corresponds to the appropriate medical DRG codes, while code B corresponds to the surgical DRG codes. It can be easily seen that these two measures behave quite differently from each other. For DRG code A, there are few treatments related to the recovery material group. On the other hand, about 25% of treatments in the basic treatment group are considered to be overtreatment. However, the picture changes completely with DRG code B. Approximately 6.0% of all treatments appear in the recovery material group. At the same time, only 2.5% of the treatments in the basic treatment group are considered to be overtreatment. On the contrary, about 12% of treatments in the recovery material group are considered to be overtreatment. As seen from the above observations, categorizing treatments into more homogeneous groups may lead to more insightful analysis.

Nevertheless, we agglomerated some of the divisions of treatments showing similar characteristics into a single category. For example, Figure 3.6 shows that the category values are too finely grained. HIRA's claim filing process requires for the associated category value to be exactly identified and entered. However, our proposed model does not require such fine granularity in terms of the categories, and it suffices to agglomerate some of the finer categories if their medical implications are similar.



| Original codes |
|---|
| Admission fee(4 bedded room) |
| Admission fee(5 bedded room) |
| Admission fee(6 bedded room) |
| Compounding fee for discharged patients (for 2days) |
| Compounding fee for discharged patients (for 3days) |
| Compounding fee for discharged patients (for 4days) |

| Integrated codes |
|---|
| Admission fee |
| Compounding fee<br>for discharged patients |

Figure 3.6: An example of unifying categories with similar meaning

### 3.4.2 Performance Measures

We compare the performance of our DRG model against the DEP model when the patient distribution changes from the training set to the test set. First, in the case of comparing models with the classification performance of treatments, we follow the process described in subsection 3.3.2.

Before we elaborate on ways to measure the proposed model's performance on classifying different claims filed, we need to draw a clean line between the normal claim and the abnormal claim. In this study, we define an abnormal claim to include more than a single overtreatment assigned or practiced as part of the claim. When we classify claims, not treatments, the problem of class imbalance aggrevates. The ratio of overtreatments to all treatments amounts only to from 0.6% to 5%, while the ratio of abnormal claims among all claims is about from 15% to 40%. In this case, we train several classification models in order to fully utilize claim-level information by employing the decision tree (DT), random forest (RF), neural network (NN), and logistic regression (LR) models.

### 3.4.3 Experimental Settings

The treatment-classification model comprise of a neural network with embedding layers accounting for the categorical variables with high cardinality. ReLU [64] is used as the activation function for the hidden layers to reflect non-linearity when drawing the decision boundary. We employ Adam [39] optimizer with the initial learning rate set at 0.0002. In order to prevent overfitting, we use dropout [85] and early stopping [74] techniques. For categorical variables with high cardinality, we tried different embedding dimensions corresponding to their cardinality. Since the problem of class imbalance eminent, we over-sample data points from the minority class for every batch.

We tested our model with a variety of hyper-parameter settings for each department and DRG code. In each case, we selected a model with the largest area under the precision-recall curve (AUPRC) reported during the validation phase [21]. Then, we selected the best threshold value with the best f1-score. Our selection process is illustrated in Figure 3.7. Pytorch package [70] was used for training the treatment-level information, while scikit-learn package [71] was employed to train to classify the claim-level information.

## 3.5 Results

### 3.5.1 Overtreatment Detection

We report the performance of the treatment classification in Table 3.2. In every case, the claim distribution is simlar between the training set and the test set 1. For the

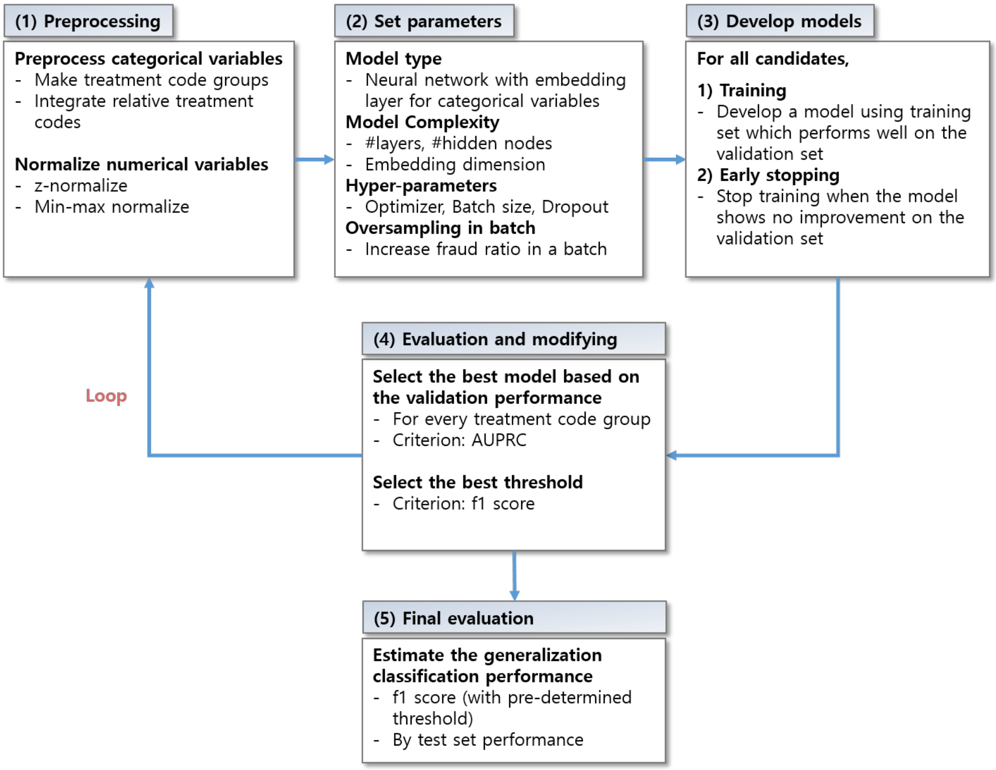

(1) Preprocessing
Preprocess categorical variables
- Make treatment code groups
- Integrate relative treatment codes
Normalize numerical variables
- z-normalize
- Min-max normalize
(2) Set parameters
Model type
- Neural network with embedding layer for categorical variables
Model Complexity
- #layers, #hidden nodes
- Embedding dimension
Hyper-parameters
- Optimizer, Batch size, Dropout
Oversampling in batch
- Increase fraud ratio in a batch
(3) Develop models
For all candidates,
1) Training
- Develop a model using training set which performs well on the validation set
2) Early stopping
- Stop training when the model shows no improvement on the validation set
Loop
(4) Evaluation and modifying
Select the best model based on the validation performance
- For every treatment code group
- Criterion: AUPRC
Select the best threshold
- Criterion: f1 score
(5) Final evaluation
Estimate the generalization classification performance
- f1 score (with pre-determined threshold)
- By test set performance

Figure 3.7: Training and selecting the best abused treatment detection model

departments C and E, the distribution is similar between the test set 1 and the test set 2. In other cases, there is a difference in the distribution between test sets.

As for test set 1, the DEP model performs slightly better than the DRG model in most cases. It can be seen that the proposed model potentially learns different types of patients better. However, for the test set 2, different results are observed. In most cases, the DRG model performs better as compared to the DEP model. Moreover, the decrease in the performance of the DEP model is quite dramatic, while the DRG model shows relatively more stable performance. So, when the distribution for most prominent patient type is shifted, the DEP model fails to perform well. However, as for the DRG models, every model trains the treatment pattern according to the each patient type. Hence, even if the patient distribution changes, the degradation in performance is not so severe. Altogether, we conclude that the DRG model is more robust to the change in the distribution of the patient type as compared to the DEP model when classifying the treatments.

Table 3.2: Performance of the overtreatment detection

| Department | Model | Accuracy | | Precision | | Recall | | F1 | |
|---|---|---|---|---|---|---|---|---|---|
| | | Test1 | Test2 | Test1 | Test2 | Test1 | Test 2 | Test1 | Test2 |
| A | *DEP* | 0.9473 | 0.9565 | 0.2132 | 0.2027 | 0.4305 | 0.2809 | **0.2852** | 0.2355 |
| | *DRG* | 0.9485 | 0.9560 | 0.2090 | 0.2197 | 0.3979 | 0.3318 | 0.2741 | **0.2644** |
| B | *DEP* | 0.9678 | 0.9769 | 0.2907 | 0.3545 | 0.4227 | 0.2501 | **0.3445** | 0.2933 |
| | *DRG* | 0.9642 | 0.9703 | 0.2575 | 0.2739 | 0.4178 | 0.3303 | 0.3186 | **0.2994** |
| C | *DEP* | 0.9423 | 0.9373 | 0.5247 | 0.5017 | 0.6382 | 0.6259 | **0.5759** | 0.5569 |
| | *DRG* | 0.9419 | 0.9403 | 0.5221 | 0.5210 | 0.6378 | 0.6472 | 0.5742 | **0.5773** |
| D | *DEP* | 0.9790 | 0.9745 | 0.1333 | 0.0347 | 0.4025 | 0.1285 | 0.2003 | 0.0546 |
| | *DRG* | 0.9901 | 0.9922 | 0.2768 | 0.2301 | 0.3227 | 0.1479 | **0.2980** | **0.1801** |
| E | *DEP* | 0.9807 | 0.9809 | 0.3340 | 0.3927 | 0.4416 | 0.3884 | **0.3803** | **0.3905** |
| | *DRG* | 0.9780 | 0.9780 | 0.2952 | 0.3381 | 0.4631 | 0.4122 | 0.3606 | 0.3715 |

### 3.5.2 Abnormal Claim Detection

Table 3.3 reports the performances of DEP models, DRG models, and the models that utilize claim-level variables. Above all, we can see that the DEP models and the DRG models exploiting the treatment-level variables perform better than the other models. It implies that the models with the treatment-level variables may perform better than the models with the claim-level variables when classifying claims.

Also, the degradation in the performance of the DEP model from the test set1 to the test set 2 is clearly apparent; yet, the decrease in performance for the DRG model is not as large as the DEP model. It may suggest that the DRG model is more robust to the changes in the distribution of the patient type than the DEP model when classifying the claims.

Table 3.3: Performance of the abnormal claim detection

| Department | Model | Accuracy | | Precision | | Recall | | F1 | |
|---|---|---|---|---|---|---|---|---|---|
| | | Test1 | Test2 | Test1 | Test2 | Test1 | Test 2 | Test1 | Test2 |
| A | *DEP* | 0.5989 | 0.6156 | 0.5013 | 0.5193 | 0.8593 | 0.7486 | **0.6332** | 0.6132 |
| | *DRG* | 0.5969 | 0.6186 | 0.4998 | 0.5205 | 0.8627 | 0.7988 | 0.6329 | **0.6306** |
| | *LR* | 0.4976 | 0.5128 | 0.4282 | 0.4402 | 0.9465 | 0.9473 | 0.5896 | 0.6011 |
| | *NN* | 0.4973 | 0.5131 | 0.4286 | 0.4408 | 0.9555 | 0.9558 | 0.5918 | 0.6034 |
| | *DT* | 0.5763 | 0.5747 | 0.4396 | 0.4451 | 0.4050 | 0.3960 | 0.4213 | 0.4191 |
| | *RF* | 0.4771 | 0.4893 | 0.4165 | 0.4267 | 0.9266 | 0.9261 | 0.5747 | 0.5843 |
| B | *DEP* | 0.6568 | 0.6709 | 0.5213 | 0.5942 | 0.7095 | 0.4940 | **0.6010** | 0.5395 |
| | *DRG* | 0.6179 | 0.6295 | 0.4847 | 0.5200 | 0.7750 | 0.6643 | 0.5964 | **0.5832** |
| | *LR* | 0.4558 | 0.4688 | 0.3967 | 0.4171 | 0.9277 | 0.9146 | 0.5558 | 0.5729 |
| | *NN* | 0.4462 | 0.4592 | 0.3939 | 0.4134 | 0.9451 | 0.9266 | 0.5561 | 0.5717 |
| | *DT* | 0.5761 | 0.5744 | 0.4178 | 0.4463 | 0.3941 | 0.3837 | 0.4056 | 0.4126 |
| | *RF* | 0.4469 | 0.4555 | 0.3932 | 0.4110 | 0.9339 | 0.9177 | 0.5534 | 0.5677 |
| C | *DEP* | 0.7085 | 0.7129 | 0.6951 | 0.7198 | 0.9207 | 0.9043 | **0.7922** | 0.8016 |
| | *DRG* | 0.7085 | 0.7200 | 0.6984 | 0.7266 | 0.9097 | 0.9035 | 0.7902 | **0.8054** |
| | *LR* | 0.5776 | 0.6069 | 0.5762 | 0.6069 | 0.9941 | 0.9934 | 0.7296 | 0.7535 |
| | *NN* | 0.5827 | 0.6126 | 0.5792 | 0.6103 | 0.9942 | 0.9937 | 0.7319 | 0.7562 |
| | *DT* | 0.6037 | 0.5968 | 0.6624 | 0.6845 | 0.6293 | 0.6181 | 0.6454 | 0.6496 |
| | *RF* | 0.6007 | 0.6293 | 0.5920 | 0.6237 | 0.9754 | 0.9755 | 0.7368 | 0.7609 |
| D | *DEP* | 0.5621 | 0.4753 | 0.2337 | 0.1918 | 0.7017 | 0.6709 | 0.3506 | 0.2983 |
| | *DRG* | 0.8038 | 0.8033 | 0.4347 | 0.3749 | 0.5480 | 0.2741 | **0.4848** | 0.3167 |
| | *LR* | 0.6381 | 0.6119 | 0.2416 | 0.2643 | 0.6180 | 0.6116 | 0.3474 | **0.3691** |
| | *NN* | 0.8443 | 0.8143 | 0.6667 | 0.3846 | 0.0020 | 0.0012 | 0.0041 | 0.0024 |
| | *DT* | 0.7619 | 0.7346 | 0.2431 | 0.2622 | 0.2497 | 0.2372 | 0.2464 | 0.2491 |
| | *RF* | 0.5295 | 0.4977 | 0.2115 | 0.2264 | 0.7398 | 0.7062 | 0.3289 | 0.3429 |
| E | *DEP* | 0.7119 | 0.6893 | 0.5293 | 0.5311 | 0.6970 | 0.6076 | **0.6017** | 0.5668 |
| | *DRG* | 0.6999 | 0.6868 | 0.5138 | 0.5266 | 0.7188 | 0.6308 | 0.5992 | **0.5740** |
| | *LR* | 0.6068 | 0.6089 | 0.3689 | 0.3914 | 0.8622 | 0.8471 | 0.5167 | 0.5354 |
| | *NN* | 0.6008 | 0.6015 | 0.3682 | 0.3894 | 0.8907 | 0.8768 | 0.5210 | 0.5393 |
| | *DT* | 0.6923 | 0.6824 | 0.3676 | 0.3946 | 0.3643 | 0.3628 | 0.3659 | 0.3780 |
| | *RF* | 0.5641 | 0.5627 | 0.3468 | 0.3675 | 0.8916 | 0.8935 | 0.4993 | 0.5208 |

## 3.6 Summary

The distribution of health insurance claims shifts from time to time due to seasonality of several diseases. Nevertheless, there are few abuse detection models which effectively account for seasonality. Most studies employ coarsely grained derived variables, defined at the provider or claim-level, for example, which makes it even more difficult to address the seasonality issues when modelling abuse detection algorithms.

In the previous chapter, we proposed an abusive provider detection model using treatment-level information. The proposed model detects abusive providers for each given department. The underlying assumption of the proposed abusive provider detection model is that claims are similarly distributed in the training and the test sets. This assumption may not hold true for some departments. If we ignore this difference in modeling, the performance will be decreased.

In order to tackle seasonality issues, we implement an abuse detection model which incorporates treatment classification to detect abuse cases for each DRG code. DRG is a type of the patient classification system (PCS), which classifies patients into groups based on clinical features and the consumption pattern of medical resources. We observe that claims with the same DRG code show similarity regardless of the timing of the filing. Instead of running a single model separately for each department, we propose to a model embodying multiple structures specific to DRG codes selected as important for each given department. We also run the single model for each department and compare the results with our proposed model. Experiment results show our proposed model performs well across different time windows, while the department-wise single models show degradation in performance.

This paper contributes to the existing literature by building the abuse detection model which effectively accounts for seasonality in health insurance claims. Moreover, we provide ground evidence for DRG, an ontology originally designed to categorize patients, to be used in the claim review process.

# Chapter 4

# Detection of overtreatment with graph embedding of disease-treatment pair

## 4.1 Background

Practitioners can prescribe a wild range of different treatments for the same patient. Moreover, there exist myriads of drugs that share the same efficacy. Yet, practitioners have a tendency to stick to their preferred choice of the drug and prescribe it to their patients, even though other options are available. The product may be selected based on the practitioners' clinical experience or personal preference. This same affinity towards specific choices can be observed not only from drug prescription but also from practicing medical procedures.

Suppose there are two practitioners who prescribe different drugs, which actually have similar medical efficacy, to the same patient. Two separate claims will be filed for each practice. Now, when the reviewers examine these claims, based on their expertise, it can easily be determined that both cases are normal since both prescriptions are appropriate responses to patient's disease. The machine, however, will have to establish such relational knowledge from scratch, and it will have to learn it from data. However, previously suggested models are not designed to efficiently deal with the complex relationships between the disease and the treatments.

In the previous chapters, embedding vectors for both diseases and treatments are learned simultaneously. Hence, these embedding vectors are in separate spaces. In order to add the relationship between disease and treatment, we simply concatenate the embedding vectors additionally feed to the model. However, it is not sufficient of an approach to include complex disease-treatment relationship.

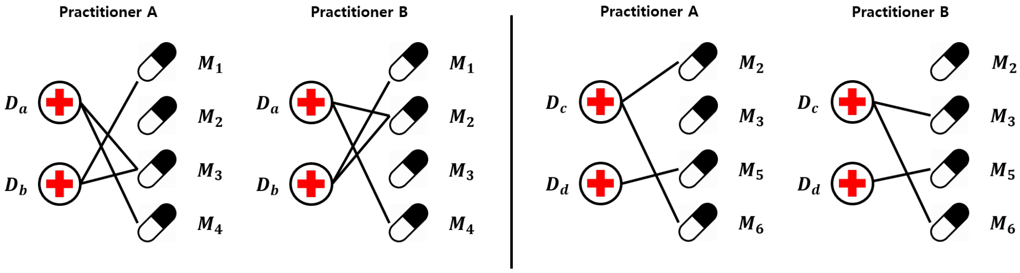


Figure 4.1: An example of different prescription from different practitioners to the same patient

Let us illustrate the reasoning behind this assertion by taking a toy example. Suppose there is a claim in the test set with the same diseases as found in some of the claims in the training set. Suppose, however, treatments prescribed in the test claim are different from those prescribed in the training claims, even though the patients in these claims suffered from similar diseases. A naïve model will classify the treatment prescribed in the test set at random, because it is a practice pattern unseen during the training. The naive model does not know that the diseases in both train and test claims are similar to each other. It won't be able to learn that, even though the prescribed treatments differ in the train claim and the test claim, the medical efficacy of the treatments are actually very similar. One the contrary, if the correct disease-treatment relationship can be modeled before the training, then the following abuse detection model would certainly perform better.

In this chapter, we propose an overtreatment detection model which considers the intricate disease-treatment relationships in prior to training. The proposed method consists of two stages. During the first stage, the disease-treatment network is constructed from the claims data. During the second stage, the model is trained to learn vector representations of entities from the disease-treatment network using node embedding methods. With the trained embedding vectors, we predict link formation between treatment and diseases in the claim in order to determine whether the treatment listed in the given claim is unnecessary to the subject patient. We test employing different network embedding models and suggest strategies to choose the most appropriate method. Our selection metric is the average performance on link prediction between the disease and the treatment.

The rest of the chapter is organized as follows. In section 4.2, we review the literature on graph embedding methods and the applications of the graph embedding method in biomedical data. Also, we introduce several studies about medical concepts embedding. Section 4.3 provides detailed descriptions of the proposed model. In section 4.4, we elaborate on experiment settings. We also describe the data in this section. Section 4.5 reports the experiment results. Finally, section 4.6 concludes the paper.

## 4.2 Literature review

In this section, we explained some state-of-the-art graph embedding network methods and their application to biomedical data. In subsection 4.2.1, we briefly introduced some graph embedding methods. In subsection 4.2.2, we reviewed about applying graph embedding methods in biomedical data. Finally, We described some methods related to medical concept embedding in section 4.2.3.

### 4.2.1 Graph embedding methods

The graph embedding methods can be divided into four categories: matrix factorization(MF)-based methods, random walk-based methods, deep learning-based methods, and other methods. In this subsection, we briefly reviewed each category and corresponding methods.

**MF-based methods**

Originally, the matrix factorization method has been widely adopted for dimension reduction of the data matrix. The data matrix is factorized into lower-dimensional matrices while preserving the manifold structure. The MF-based graph embedding method is factorizing matrices, which represent graph properties, to obtain node embedding vectors in lower dimension space. There are several graph embedding methods that utilize matrix factorization methods such as locally linear embedding(LLE) [76], Laplacian eigenmaps(LE) [7], Graph Factorization(GF) [2], GraRep [9], and HOPE [69].

In LLE, find k-nearest neighbors(k-NN) of each data and make an adjacency matrix based on the k-NN result. Then, factorize the matrix using the matrix factorization method such as Singular Value Decomposition(SVD). While LLE [76] uses the constructed matrix itself, LE [7] factorizes graph Laplacian Eigenmaps to preserve pairwise node similarities. It converts finding embedding vector problem to generalized eigenvector problem. GF [2] directly factorize the proximity matrix of a graph under each edge is already existed. These methods aim to preserve 1st-order proximity. However, many networks have important features in high-order proximities.

GraRep and HOPE are two important methods that preserve high-order proximities. In GraRep [9] method, authors capture network the local and global structure by generating multiple $k$-step embedding vectors by factorizing multiple $k$-step transition probability matrices, and concatenate those vectors. HOPE [69] is used to get embedding vectors of the directed graph which has asymmetric transitivity. The basic idea of HOPE is that a node should have two different embedding vectors because each node can be used as a target node. It defines some important high-order proximity measures such as Katz index [37], and get embedding vectors that preserve such measures. SVD is used to factorize the matrices in both methods.

**Random walk-based methods**

Random walk is a stochastic process with random variables $W^1_{v_i}$,$W^2_{v_i}$,...,$W^k_{v_i}$ such that every value is randomly chosen from the neighbors of previous value. In other words, if $W^j_{v_i} = v_j$, then $W^j_{v_i}$ must be randomly chosen from $N(v_j)$, which means the neighbors of node $v_j$. In short, a random walk in a network is a node sequence in which every node is connected to the previous node. It is commonly used to capture structural relationship between nodes of the network. Perozzi et al. [72] found that the distribution of vertices appearing in short random walks is similar to the distribution of words appearing in sentences under certain circumstances. Inspired by this observation, they suggested a method named DeepWalk, which utilizes SkipGram [61] model in random walks to learn the embedding vector of each node. Also, the hierarchical softmax method was used to train SkipGram model ([62], [63]). After this study, there have been several papers that utilize the word embedding model in

NLP to random walks. Grover and Leskovec [29] suggested node2vec model that uses the biased random walk rather than unbiased random walk in DeepWalk. They used the biased random walk to preserve the local structure and the global structure by using breadth-first searching and depth-first searching in generating random walks. With these random walks, they trained SkipGram model with negative sampling. Perozzi et al. [73] proposed Walklets, which is another extension of DeepWalk. They modified a strategy of generating random walk to skipping some nodes each random walk. By this strategy, they made it possible to generate random walks that contain multiple $k$-steps proximities.

Diffusion component analysis (DCA) [13] is another random walk-based embedding method, but quite different from previous methods. While previous methods are node embedding methods utilizing SkipGram model, DCA calculates the diffusion state that is defined as the probability distribution in stationary state with random walk with restart (RWR) strategy. This strategy captures both local and global structural property. Also, it makes possible to overcome the noise and sparsity of the network, so that this method can be used in the biological network.

While these methods were concerned only about proximities, struc2vec [75] is a graph embedding method that preserves structural identity. The authors of struc2vec explain the structural identity as *a concept a symmetry in which network nodes are identified according to the network structure and their relationship to other nodes* [75]. In other words, a node pair having structural identity means both nodes perform similar roles in the network. Firstly, define the structural similarity and construct a multilayer weighted network where all nodes exist in every layer. Then, generate the context for each node by using biased random walks with the multilayer network.

Then, train a SkipGram model with the hierarchical softmax method to learn node embedding vector for each node with random walks.

These methods are for the homogeneous networks, which refer to networks with a single type of nodes and edges. However, there are much more networks which are not homogeneous such as author-paper-venue, customer-products-seller network. These networks are called heterogeneous networks which include different types of nodes and edges. There are several studies about embedding methods for these networks, such as metapath2vec [23] and Heterogeneous Information Network Embedding (HINE) [35]. Here, both methods are random walk-based method. Except for generating random walks method, metapath2vec is quite similar to DeepWalk. They suggest meta-path based random walks for the heterogeneous network, which generates random walks by pre-defined node type sequence. Otherwise, HINE [35] does not utilize SkipGram method in training. The authors first defined two meta-path based proximity measures for a heterogeneous network. Then, train embedding vectors of nodes while preserving those proximities.

**Deep learning-based methods**

Deep learning has been achieved success in various domains. Deep learning-based embedding methods are the embedding methods that utilize some deep learning architectures. SDNE [92] is a kind of node embedding model which utilizes the deep auto-encoder to proximity matrix of the network to map it to nonlinear latent space while preserving the network structure. By using the auto-encoder, the embedding vector preserves the second-order proximity. It makes 1st order proximity also be preserved by applying Laplacian eigenvector proximity measure to embedding vectors. DNGR [10] is another model that utilizes auto-encoder structure The authors chose the stacked denoising auto-encoder structure to find non-linear embedding vectors in low dimensional space and robust to the noise of the network. Graph convolutional network (GCN) [41] and variational graph auto-encoder (GAE) [40] are also important deep learning-based model. Both of them use the convolutional neural network (CNN) in network data which achieves great success in the computer vision domain. GCN applies the convolutional operation to network data by using the proximity matrix and feature matrix of the network. GAE is a kind of auto-encoder that uses GCN encoder and the simple inner product decoder.

**Other methods**

There are several important methods that are not included in any category. Multidimensional scaling(MDS) [33] learns embedding vectors by preserving the distance of all node pairs in the embedding space. However, it does not consider different relationships might have different importance. Isomap [88] overcome this shortage by constructing k-NN network and learn embedding vectors while preserving the distance between a node and its k-NNs. LINE [86] is a node embedding method that preserves the first-order and second-order proximity. The authors suggested preserving 1st order proximity by minimizing the distance between the empirical distribution of nodes in the original graph and the distribution from embedding space. Also, they suggest minimizing the distance between the empirical conditional distribution of 'context' node $v_j$ given a single node $v_i$ and the conditional distribution of them in the embedding space.

Table 4.1: graph embedding methods

| Category | Algorithm | Method |
|---|---|---|
| Matrix factorization | LLE [76] | matrix factorization (e.g. SVD) |
| | LE [7] | matrix factorization (e.g. eigen-decomposition) |
| | GF [2] | matrix factorization (e.g. SVD) |
| | GraRep [9] | matrix factorization (e.g. SVD) |
| | HOPE [69] | matrix factorization (e.g. SVD) |
| Random walk | DeepWalk [72] | skip-gram with random walk |
| | node2vec [29] | skip-gram with random walk |
| | Walklets [73] | skip-gram with random walk |
| | DCA [13] | stationary distribution with random walk with restart strategy |
| | struc2vec [75] | skip-gram with random walk |
| | metapath2vec [23] | skip-gram with meta-path based random walk |
| | HINE [35] | proximity preserving model with meta-path based random walks |
| Deep learning | SDNE [92] | Autoencoder with proximity matrix |
| | DNGR [10] | Denoising autoencoder with PPMI matrix |
| | GCN [41] | CNN model with adjacency matrix and feature matrix |
| | GAE [40] | Autoencoder with GCN encoder and simple inner product decoder |
| Others | MDS [33] | Preserving Euclidean distances of all node pairs |
| | Isomap [88] | Preserving Euclidean distances of each node and its $k$-nearest neighbors |
| | LINE [86] | Preserving 1st-order and 2nd-order proximity |

### 4.2.2 Application of graph embedding methods to biomedical data analysis

The network embedding method is applied mainly in three topics: pharmaceutical data analysis, multi-omics data analysis, and clinical data analysis. In this subsection, we explained each category and review several studies.

**Pharmaceutical data analysis**

The usage of graph embedding or graph analysis in pharmaceutical data can be categorized as three important issues: drug-target interaction (DTI) prediction, drug-drug interaction (DDI) prediction, and drug-disease association (DDA) prediction. DTI prediction means predicting the interactions between drugs (chemical compound) and target (protein). DDI prediction is to predict the result of drug co-prescription. DDA prediction means predicting the clinical result when a patient, who has a specific disease, takes a specific drug.

Previously, DTI prediction was mainly performed by constructing proximity matrices and factorize them by matrix factorization methods. Yamanashi et al. [102] proposed a method of predicting unknown DTI by using known DTI data, chemical data, and genomic data. They construct a known drug-target bipartite network by DTI data and factorize the similarity matrix by eigenvalue decomposition. Next, train models that represent the correlation between embedding space and chemical/genomic space. Then, the unknown DTI can be inferred by the model. Cobanoglu et al. [18] proposed a method that predicting DTI by using the collaborative filtering method only with known DTI data, not any external data. They applied the probabilistic matrix factorization method to the known DTI network to get embedding vector of each node in drug-protein and predict unknown DTI by active learning with learned embedding vectors. Ezzat et al. [25] suggested a method that predicts DTI by the graph embedding method and ensemble learning. They conducted a feature sub-spacing to inject diversity for classifier ensemble and tried three different dimension reduction methods: SVD, Partial Least Squares(PLS), and LE. Then, train homogeneous base learners with the resulting vectors and predict with each model's score. Also, there is another method that uses the k-NN method and graph regularization matrix factorization method to predict unknown DTI [26].

While MF-based methods were used in previous studies, random-walk based methods are also commonly used in DTI prediction. Luo et al. [54] developed a model named DTINet to predict DTIs from a heterogeneous network that is constructed by integrating drug-related information. They used extended DCA to learn embedding vectors for each node in the heterogeneous network. Then find the best projection from drug space to target space by finding mapped feature vectors of drugs are similar to the known interacting target. Then, infer new interactions of a drug by ranking the target candidates and projected feature vector of the drug. Zong et al. [110] proposed a similarity-based DTI prediction method by constructing a drug-target-disease tripartite network. After construction, train embedding vectors for each node to predict the drug-target association. They utilized DeepWalk method to learn embedding vectors. Alshahrani et al. [3] proposed another method that integrates external information to construct a heterogeneous network. They integrated gene ontology(GO), protein-protein interactions(PPIs), DTIs, gene-disease interactions, drug side effects, and disease-phenotype information to construct the network. They utilized a modified DeepWalk method to learn embedding vectors that captures the structure of the network. Then, they trained the logistic regression model to predict the unknown DTIs.

There were also several studies predicting DDIs. Zhang et al. [107] proposed a method that formulates DDI prediction as a matrix completion problem. Firstly, they integrated multiple external drug-related information and learned embedding vectors. Also, they suggested a method named 'Manifold Regularized Matrix Factorization' (MRMF), which is a kind of MF-based embedding method, to learn embedding vectors. Then, they found similarity factors between node pairs with embedding vectors and known DDI information. Ma et al. [56] proposed a model that calculates similarities between drugs in multi-view. They used GAE to integrate multiple types of drug features and attentive model to make the model adaptive to data. They also used the model to predict unknown DDI. Zitnik et al. [109] proposed a model named 'Decagon', which is aimed to predict DDI, especially polypharmacy side effects. Firstly, they constructed a multimodal graph from PPIs, DTIs, and polypharmacy side effect information. Different types of interactions are labeled by different edge types. The unknown DDIs are predicted by link prediction between drug nodes using modified GAE. The node information is encoded by the GCN based encoder. Then, the decoder takes pairs of embedding vector and scores the edge between them.

Predicting DDAs is also an important issue in pharmaceutical data analysis. Dai et al. [20] first embedded gene-gene interaction network by eigenvalue decomposition and get embedding vectors of drugs and disease with the gene embedding vectors, drug-gene interactions, and disease-gene interactions. Then, factorize the known drug-disease association matrix. Finally, the unknown DDAs can be inferred by the embedding vectors of drugs and diseases, and the matrix factorization result of known drug-disease association matrix. Wang et al. [94] constructed a drug-disease network from free text, especially extracted from papers in PubMed and learn embedding vector with modified LINE. Then, the correlation of the drug-disease pair is calculated by the embedding vectors and find DDA patterns.

**Multi-omics data analysis**

The term ‘omics’ means a field of study in biology that ends with ‘-omics’, such as genomics. These studies are about researching characteristics of biological molecules such as their structures, functions, or dynamics. The network-based approach is a valuable method in these studies in finding a relationship between entities. Here, we reviewed three important topics that utilize graph embedding methods: proteomics, genomics, and transcriptomics data analysis.

Many studies that apply graph embedding methods in proteomics is focused on assessing and predicting PPIs, or predicting protein functions. Kuchaiev et al. [45] proposed a de-noising PPIs model with MDS-based graph embedding approach to address high false positive and false negative in PPIs. You et al. [104] used isomap to embed the PPI network in low dimensional space. Then, assess and predict the PPIs by comparing embedding vectors of the node pair. Lei et al. [49] proposed a two-step model that assesses and predicts PPIs. First, combine multiple genomic and proteomics information by logistic regression approach to construct a weighted PPI network. Then, get embedding vectors by extended isomap and predict the unknown PPIs. Wang et al. [97] proposed ProsNet, which predicts the PPI by constructing a heterogeneous molecular network and embedding the network in low dimensional space. The heterogeneous molecular network is constructed by including the molecular networks of several species and gene ontology graph. Then, the embedding vectors are calculated by meta-path based extended DCA.

Graph embedding methods are utilized for various purposes in analyzing genomic data. As We already reviewed in the previous subsection, Cho et al. [13] proposed DCA, which is an important graph embedding method, to learn node embedding vectors with RWR strategy. Wang et al. [95] proposed a method named clusDCA, which predicts the gene function. They learned embedding vectors from gene-gene interaction network and GO by DCA. Then, they trained a projection model from gene space to GO space. With projected vectors and embedding vectors in GO space, they predicted the gene function of the gene. There is also another DCA-based model named PACER that aims to pathway identification [96]. The main idea of this method is to construct a heterogeneous network and embedded gene and pathway in a unified space. They used gene expression, drug response-gene expression, PPIs, and pathway information to construct the network. Li et al. [51] proposed a model named SCRL, which aims to learn the representation of a single cell RNA sequence. The basic idea of this model is constructing cell-ContexGene and Gene-ContextGene networks and learning embedding vectors by extended LINE. Zeng et al. [106] constructed a heterogeneous gene-disease network from human genes and other species' genes information. Then, they calculated embedding vectors by factorizing the matrix and predicted the pathogenic human genes.

Transcriptomics is a study of an organism's transcriptome, which is all about RNA transcript. In this field of study, The graph embedding methods are mainly used to identify the miRNA-disease association. Shen et al. [81] developed Collaborative Matrix Factorization for miRNA-Disease Association(CMFRDA) that identifies the miRNA-disease association. They constructed a miRNA-disease bipartite graph and factorized the matrix by the SVD for initialization. Then, they update the factorized matrix until the predefined loss is converged. Li et al. [50] proposed a similarity-based miRNA-disease prediction model. They constructed the miRNA-disease bipartite network and learned similarity by embedding the network with DeepWalk. Then, they infer the miRNA-disease interaction by the distance between embedding vectors of a node pair.

**Clinical data analysis**

There are several papers about analyzing the clinical data, such as medical knowledge graph, electronic health records (EHRs) and electronic medical records (EMRs). Choi et al. [16] suggested learning embedding vectors from three different data sources: medical journals, medical claims, and clinical narratives. Different types of concepts are embedded in a common low-dimensional space. They tried two embedding methods: SkipGram and matrix factorization. Wang et al. [93] suggested a method to recommend appropriate medicine for patients. They constructed heterogeneous network by combining medical knowledge network, patient-medicine network, and patient-disease network. They trained embedding vectors of the network by using translation-based embedding method and LINE. Choi et al. [15] developed a model named GRAM, which aims to learn low-dimensional representation with medical concept ontology. They utilized the attention method to leverage the parent-child relationship of the ontology.

Table 4.2: Applications of graph embedding methods in biomedical data analysis

| Tasks | Authors | Purpose | Embedding method |
|---|---|---|---|
| Pharmaceutical data analysis | Yamanashi et al. [102] | DTI prediction | Matrix factorization |
| | Cobanoglu et al. [18] | DTI prediction | Probabilistic matrix factorization |
| | Zheng et al. [108] | DTI prediction | Matrix factorization |
| | Ezzat et al. [25] | DTI prediction | Matrix factorization |
| | Ezzat et al. [26] | DTI prediction | Matrix factorization (SVD, PLS, LE) |
| | Luo et al. [54] | DTI prediction | DCA |
| | Zong et al. [110] | DTI prediction | DeepWalk |
| | Alshahrani et al. [3] | DTI prediction | Modified DeepWalk |
| | Zhang et al. [107] | DDI prediction | matrix factorization |
| | Ma et al. [56] | DDI prediction | GAE |
| | Zitnik et al. [109] | DDI prediction | modified GAE |
| | Dai et al. [20] | DDA prediction | Eigenvalue decomposition, Matrix factorization |
| | Wang et al. [94] | DDA prediction | modified LINE |
| Multi-omics data analysis | Kuchaiev et al. [45] | Denoising PPI | Extended MDS |

Table 4.2: Applications of graph embedding methods in biomedical data analysis

| Tasks | Authors | Purpose | Embedding method |
|---|---|---|---|
| | You et al. [104] | Assessing PPI, PPI prediction | Isomap |
| | Lei et al. [49] | Assessing PPI, PPI prediction | Extended Isomap |
| | Wang et al. [97] | PPI prediction | Meta-path based extended DCA |
| | Cho et al. [13] | Node embedding in biological network | DCA |
| | Wang et al. [95] | Gene function prediction | Extended DCA |
| | Li et al. [51] | Learn the representation of single cell RNA-seq | Extended LINE |
| | Zeng et al. [106] | Predict pathogenic human genes | Matrix factorization |
| | Wang et al. [96] | Pathway identification | DCA |
| | Shen et al. [81] | Identify potential miRNA-disease association | Matrix factorization |

Table 4.2: Applications of graph embedding methods in biomedical data analysis

| Tasks | Authors | Purpose | Embedding method |
|---|---|---|---|
| | Li et al. [50] | miRNA-disease prediction | DeepWalk |
| Clinical data analysis | Choi et al. [16] | Medical concept embedding | SVD,<br>SkipGram |
| | Choi et al. [15] | Medical concept embedding | GRAM |
| | Wang et al. [93] | Medicine Recommendation | Translation based,<br>LINE |

### 4.2.3 Medical concept embedding methods

In order to apply various machine learning methods in clinical data, the medical concept in the clinical data should be vectorized. There have been researches to embedding medical concept to get embedding vector for various purpose, such as predicting patients' visits.

Choi et al. [14] proposed a medical concept representation method named med2vec from EHR datasets. Here, they define a visit vector $V_t$, and represent it as a binary vector $x_t \in 0, 1^{|C|}$ , where the $i$-th entry is 1 only if $c_i \in V_t$. Then, represent the binary vector in intermediate low dimensional space, and concatenate the vector with demographic information. Embed the concatenated vectors into the final low dimensional space and predict the other binary vectors in a context window. Not only they used inter-visit information, but also they used inter-visit information to preserve code-level information. Another work done by the Choi's team is GRAM [15], which is we already reviewed in the subsection 4.2.2. It utilized the attention model to each node in the medical concept ontology to learn low-dimensional embedding vectors that leverage the parent-child relationship. Song et al. [83] proposed another ontology-based medical concept model named MMORE. The model learns multiple embedding vectors for the ancestors of the leaf nodes and the final embedding vectors are calculated by combining those embedding vectors with an attention mechanism.

Cai et al. [8] proposed an embedding method that considers the temporal information because the scopes medical concept varies greatly in terms of temporal scope. The embedding vectors are calculated from other EMR data codes that are in a certain time window with the attention model. Xiang et al. [101] claimed that the embedding vectors of medical concepts should consider temporal dependency. They tried word2vec, PPMI, and FastText [36] with large EHR datasets to learn embedding vectors of medical concepts to overcome this issue.

## 4.3 Proposed method

This section presents our overtreatment detection model using graph embedding method. Subsection 4.3.1 details the process of the medical information network from the medical treatments found in healthcare insurance claims. More specifically, we present how to compute the edges of the network from the treatment data. Subsection 4.3.2 describe our strategies for choosing the best method for embedding the constructed network in order to carry out the overtreatment detection task. We solve the link prediction problem and compare performances among the select methods.

Finally, we construct the overtreatment detection model by using graph embedding in subsection 4.3.3. We sample negative edges from the constructed network, in particular, and use the embedding vectors of the nodes, trained and chosen as described in subsection 4.3.2, to predict links.

### 4.3.1 Network construction

Our healthcare insurance claim data consists of three parts: (1) basic claim information; (2) disease information, and; (3) treatment information. Basic claim includes claim-wise elements as claim identifiers, general practitioner (GP) information, subject patient profiles, as well as the relevant DRG code. Disease information reports the list of diseases the subject patient has. The treatment information encompass all the details of treatments that the general practitioner has prescribed the patient. Figure 2.5 illustrates an example of a claim typically found in our data set

The toughest challenge in constructing a network from in insurance claim data set is that the casual relationship between the disease and the treatment is not apparent. A claim contains information on the main and sub-diseases diagnosed as well as the type and amount of treatments, yet it still remains in dark exactly for which disease each treatment was prescribed. The absence of exact disease-treatment matching poses as a problem in the following sense: suppose an edge between a disease and a treatment is formed if they appear in the same claim. For example, if diseases A and B are listed together with treatments C and D for an arbitrary claim case c, then an edge will be formed between disease A and treatments B and C. Now, suppose that, in reality, treatment D was prescribed for disease B only and, likewise, treatment C only for disease A. Then, the edge between disease A and treatment D carries wrong relational information. That is, in other words, edge formation based on co-occurrence may lead to misleading representations.

In order to address above issue, we resort to the concept of the relative risk (RR) as a vehicle to infer the relationship and form edges accordingly ([60], [100]). Relative risk is a statistical measure of statistical method utilized in cohort studies to infer the association between an outcome and a factor.

For example, suppose ‘B’ is an outcome and ‘A’ is a factor. Then, RR(A,B) is defined as follows:

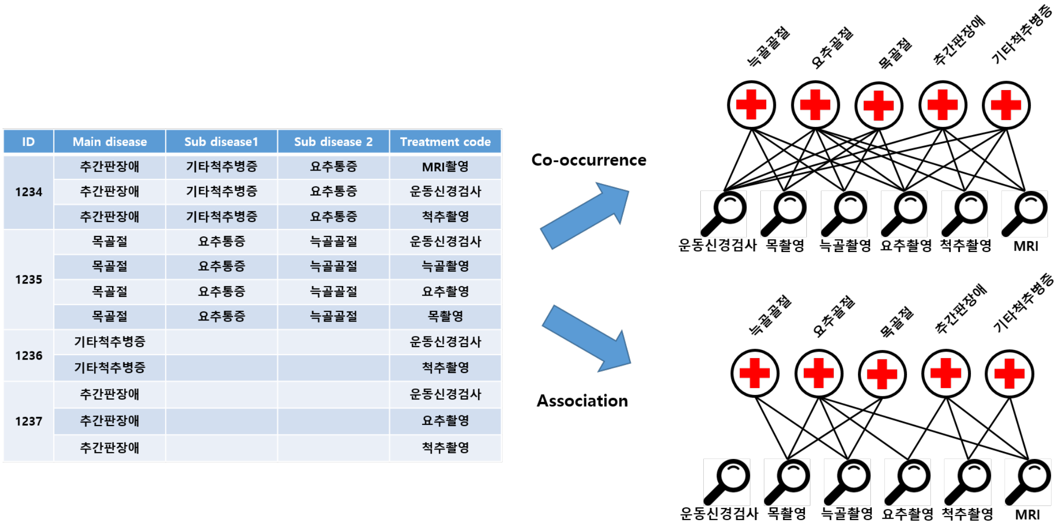
$$RR(A, B) = \frac{p(B|A)}{p(B| \sim A)}$$

If the resulting value is larger than 1, then factor ‘A’ is considered to be associated with outcome ‘B’.

Now, we construct disease-treatment network as following. We begin by forming edges between the main diseases and the RDRG codes. If a claim has a specific RDRG code and main disease(s) listed, then edges are formed between the code and the corresponding main diseases. The main diseases, then, are connected with sub-diseases listed for the same claim, if any. Finally, we form edges between the diseases and treatments by exploiting the RR measure. More specifically, RRs are computed for all the disease-treatment pairs in the training set. Then, an edge formed for the disease-treatment pair whose RR value is greater than 1. The resulting network comprises undirected, unweighted edges.

### 4.3.2 Link Prediction between the Disease and the Treatment

In order to detect overtreatment by using node embedding vectors, we first need to select the most appropriate node embedding method which learns to represent



| ID | Main disease | Sub disease1 | Sub disease 2 | Treatment code |
|---|---|---|---|---|
| 1234 | 추간판장애 | 기타척추병증 | 요추통증 | MRI촬영 |
|  | 추간판장애 | 기타척추병증 | 요추통증 | 운동신경검사 |
|  | 추간판장애 | 기타척추병증 | 요추통증 | 척추촬영 |
| 1235 | 목골절 | 요추통증 | 늑골골절 | 운동신경검사 |
|  | 목골절 | 요추통증 | 늑골골절 | 늑골촬영 |
|  | 목골절 | 요추통증 | 늑골골절 | 요추촬영 |
|  | 목골절 | 요추통증 | 늑골골절 | 목촬영 |
| 1236 | 기타척추병증 |  |  | 운동신경검사 |
|  | 기타척추병증 |  |  | 척추촬영 |
| 1237 | 추간판장애 |  |  | 운동신경검사 |
|  | 추간판장애 |  |  | 요추촬영 |
|  | 추간판장애 |  |  | 척추촬영 |

Figure 4.2: Network construction by co-occurrence and association relationship

nodes effectively from the constructed network as vectors on the embedding space. In this subsection, we detail the process of choosing the best embedding method among other candidates, which constitutes two main steps: edge sampling and link prediction.

### Edge sampling

We begin by spliting the edge set from the original network $G = (V, E)$ into two sub-graphs: the training set, $G^{trn} = (V, E^{trn})$ and the test set, $G^{tst} = (V, E^{tst})$. Since $E^{trn}$ represents all the observable, hence positive, samples, we re-denote $E^{trn}$ as $E^{trn}_{pos}$. In contrast, negative samples are not directly observed from the claims data. Thereupon, we define negative edges as the set of all the combination pairs between the diseases and treatments in the training set that are not in $E^{trn}_{pos}$. Then, sample several negative edges from this set. The number of negative edges sampled should be equal to $\left|E_{pos}^{trn}\right|$. This set of sampled negative edges are denoted by $E_{neg}^{trn}$. Similarly, we define the set of edges observed in the test set, $E^{tst}$, as the positive edge samples and denote them by $E_{pos}^{tst}$. Negative edges are sampled in a similar fashion as explained above for the training set negative samples, which is denoted by $E_{neg}^{tst}$. This validates that the set of negative edges sampled for the test set will not intersect with those in $E_{pos}^{trn}$, $E_{neg}^{trn}$, nor in $E_{pos}^{tst}$.

**Link prediction**

At this stage, we employ a selection of node embedding methods to learn to represent nodes from $G^{trn}$ as vectors on an embedding space. Suppose an arbitrary embedding model learns to present nodes $u, v$ in the $G^{trn}$, which are connected by the edge $e = (u, v)$. We denote the corresponding embedding vectors for the nodes $u, v$ by $\boldsymbol{x_u}, \boldsymbol{x_v}$, respectively. The embedding vector for the corresponding edge $e$ is defined as $\boldsymbol{x_e} = [\boldsymbol{x_u}, \boldsymbol{x_v}]$, which results from concatenating the embedding vectors of the connected nodes. We denote the sets of the embedding vectors of the edges in $E_{pos}^{trn}, E_{neg}^{trn}$ by $X_{pos}^{trn}, X_{neg}^{trn}$, respectively. Similarly, the sets of embedding vectors of edges in $E_{pos}^{tst}, E_{neg}^{tst}$ are denoted by $X_{pos}^{tst}, X_{neg}^{tst}$, respectively. Then, using $X_{pos}^{trn}, X_{neg}^{trn}$, we train a logistic regression model, $h_\theta(e)$, using to learn to classify whether a given edge is positive or negative. Finally, we evaluate the classification result of $h_\theta(e)$ with $X_{pos}^{tst}, X_{neg}^{tst}$.

The entire process for disease-treatment link prediction is illustrated in Figure 4.3. We repeat this process several times and compute the average performance of select embedding methods in solving the link prediction task. Details on the models employed in the experiment can be found in section 4.4.

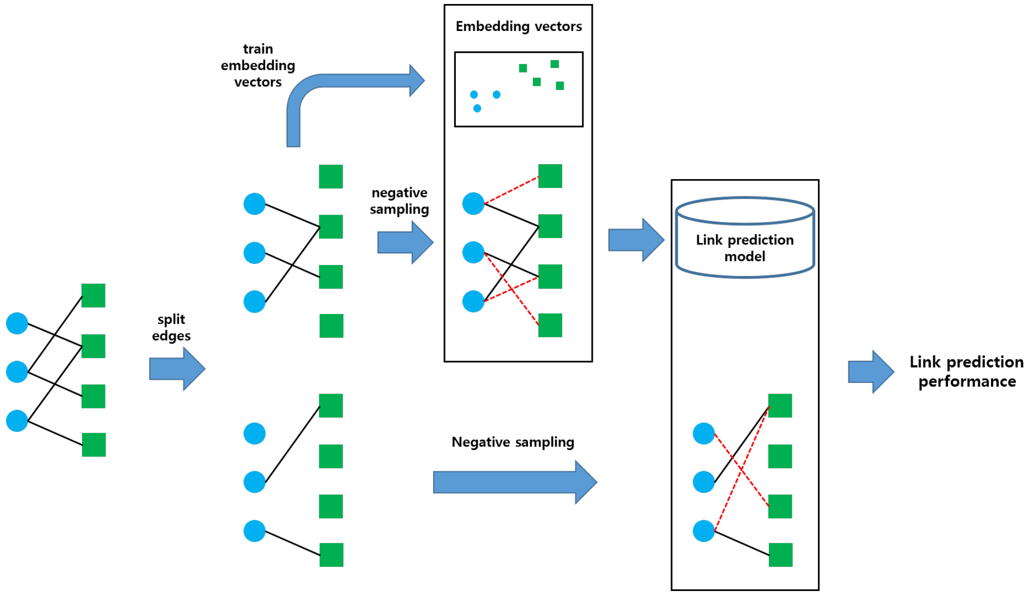


Figure 4.3: The process of link prediction between the disease and the treatment

### 4.3.3 Overtreatment Detection

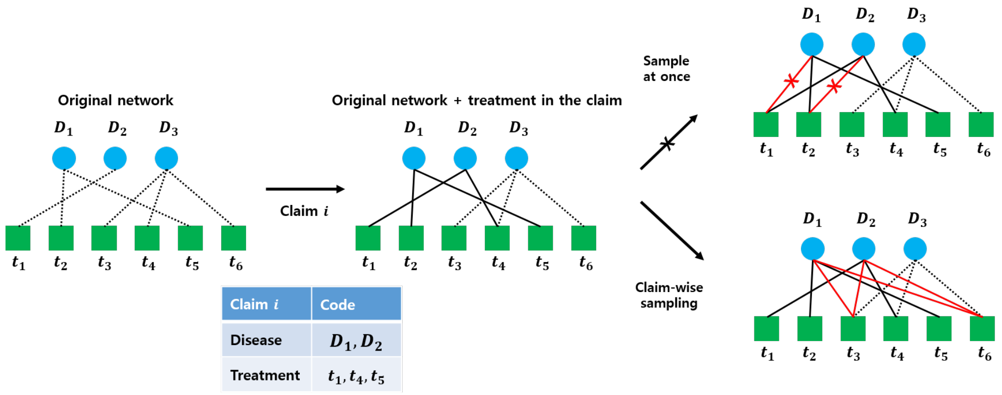
In the previous subsection, we detailed out the process for choosing the best network embedding model by comparing the average performance in disease-treatment link prediction. In this subsection, we elaborate on the framework of our overtreatment detection model which utilizes the embedding vectors of nodes. Overtreatment detection model involves two stages: edge sampling and overtreatment detection.

**Edge sampling**

Different from the previous subsection, We define the network corresponding to the training set as $G^{trn} = (V, E^{trn})$; the network corresponding to the test set, $G^{tst} = (V, E^{tst})$. Note that the nodes that do not appear during the training were also removed from the test set, hence the node set for the training is exactly what is used for the test set. The negative edge sampling process must be differentiated from what was defined in the previous subsection so as to reflect that, at this time, we need disease-treatment pair samples based on association. Take, for an example, a claim $i$ which includes diseases $d_{i1}^{trn}, d_{i2}^{trn}$ and treatment $t_{i1}^{trn}$. It may be the case that the prescription of $t_{i1}^{trn}$ is due to $d_{i1}^{trn}$, while $d_{i2}^{trn}$ is irrelevant. Such a relation should be translated to an edge $e_1 = (d_{i1}^{trn}, t_{i1}^{trn})$. On the other hand, an edge $e_2 = (d_{i2}^{trn}, t_{i1}^{trn})$ would provide misleading information, hence should not be formed. In case of subsection 4.3.2, the challenge is not as severe since $e_2$ can be sampled as a negative edge while learning to classify the disease-treatment relationship. However, in terms of evaluating the claims for overtreatment, edge sampling based on co-occurrence leads to a serious problem, since co-occurrence does not necessarily imply association. Mistakenly connecting $t_{i1}^{trn}$ to $d_{i2}^{trn}$ may lead to mis-labeling the claim $i$ as an overtreatment case, while it actually is not, for $t_{i1}^{trn}$ was an appropriate choice of prescription in response to $d_{i1}^{trn}$. Given that our ultimate goal is to detect overtreatment, such mis-labeling problem will cause grave degradation of our detection model.

In order to tackle this issue, we sample negative edges claim-by-claim, unlike the edge sampling described in the previous subsection where edges were sampled from the entire network all at once. Our negative edge sampling process preceeding the overtreatment detection proceeds as follows. Given a single claim, we identify all the diseases included in the claim. Then, we look up treatments, from the rest of the claim data set, that are not matched with either of the identified diseases. All possible combinations of the identified diseases (from the subject claim case) and the looked up treatments (from the rest of the claims data) are considered as negative edges candidates of the given claim. sample as many negative edges as the number of positive edges found in the claim.

We describe the process with an example. Suppose a claim includes diseases of $d_{i1}^{trn}, d_{i2}^{trn}$. All the treatments that are related to $d_{i1}^{trn}$ and those to $d_{i2}^{trn}$ are represented as $N(d_{i1}^{trn}), N(d_{i2}^{trn})$, respectively. Then, the negative edges are sampled from the set $E_i^{neg} = \{(u,v)|u \in \{d_{i1}^{trn}, d_{i2}^{trn}\}, v \notin \{N(d_{i1}^{trn}) \cup N(d_{i2}^{trn})\}$ where $u$ represents the disease node and $v$, the treatment node. Here, treatment nodes matched with the disease nodes from corresponding to claim $i$ is not in fact related to any of the given diseases. We denote the edge set from the training set as $E_{pos}^{trn}$; those from the test set, $E_{pos}^{tst}$. Similarly, the negative disease-treatment edge set that sampled for the disease nodes found in the training set is denoted by $E_{neg}^{trn}$; for those found in the test set, $E_{neg}^{tst}$.



| Claim $i$ | Code |
|---|---|
| Disease | $D_1, D_2$ |
| Treatment | $t_1, t_4, t_5$ |

Figure 4.4: Claim-wise negative edge sampling

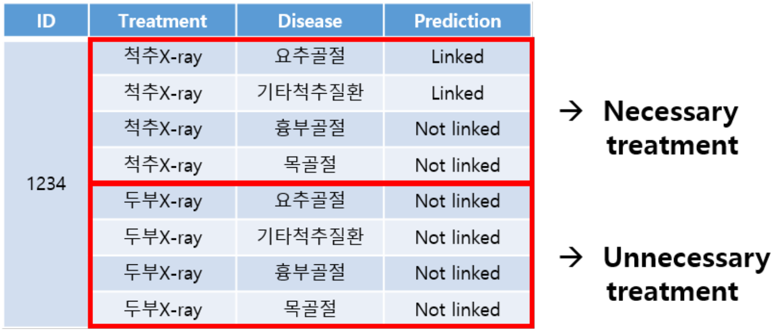
### Overtreatment detection

We propose to detect overtreatment in two different ways. First approach is to detect overtreatment naively, using the network resulting from the training set, per se. Another approach involves training node embedding models. Both approaches assume that a given case is associated with overtreatment if the link between the diseases and the treatments listed in the claim. The naïve network approach proceeds as follows: if there is a disease-treatment edge in $E_{pos}^{tst}$ which does not exists in $E_{pos}^{trn}$, the corresponding treatment is classified as overtreatment. For example, suppose that a claim includes diseases $d_1, d_2, \ldots, d_m$ and a treatment $t_k$ in the test set. If the treatment satisfies the condition of $(d_1, t_k), (d_2, t_k), \ldots, (d_m, t_k) \notin E_{pos}^{trn}$, we classify the subject treatment as overtreatment.

In case of node embedding approach, we first train our model to learn to vector representations of nodes that are connected by the edges in $E_{pos}^{trn}$. Then, a logistic regression model $h_\theta(e)$ is employed to learn to classify edges using $E_{pos}^{trn}$ and $E_{neg}^{trn}$. We test the training results using $E_{pos}^{tst}$ and $E_{neg}^{tst}$. If the test result reports that all of the diseases are not connected to a treatment found in the given claim, and the corresponding treatment is considered as overtreatment. Mathematically, we denote the embedding vectors of diseases $d_1, d_2, \ldots, d_m$,by $\boldsymbol{x_{d_1}}, \boldsymbol{x_{d_2}}, \ldots, \boldsymbol{x_{d_m}}$, and the embedding vector of the treatment $t_k$ by $\boldsymbol{x_{t_k}}$. Given the treatment $t_k$, if the prediction results for each respective diseases included in a given claim, $h_\theta([\boldsymbol{x_{d_1}}, \boldsymbol{x_{t_k}}]), h_\theta([\boldsymbol{x_{d_2}}, \boldsymbol{x_{t_k}}]), \ldots, h_\theta([\boldsymbol{x_{d_m}}, \boldsymbol{x_{t_k}}])$ are all negative, then treatment $t_k$ is classified as overtreatment. We graphically illustrate the overall framework of our overtreatment detection model in Figure 4.6.

## 4.4 Experiments

In order to evaluate our proposed model, we experiment on real-world data. Our dataset consists of health insurance claims submitted to HIRA in 2017. Subsection



| ID | Treatment | Disease | Prediction |
|---|---|---|---|
| 1234 | 척추X-ray | 요추골절 | Linked |
| | 척추X-ray | 기타척추질환 | Linked |
| | 척추X-ray | 흉부골절 | Not linked |
| | 척추X-ray | 목골절 | Not linked |
| | 두부X-ray | 요추골절 | Not linked |
| | 두부X-ray | 기타척추질환 | Not linked |
| | 두부X-ray | 흉부골절 | Not linked |
| | 두부X-ray | 목골절 | Not linked |

→ Necessary treatment
→ Unnecessary treatment

Figure 4.5: Unnecessary treatment detection by link prediction result

4.4.1 provides detailed descriptions of our dataset. Subsection 4.4.2 presents the training details.

### 4.4.1 Data Description

As described in subsection 2.4.1, there are several databases separately stored within the HIRA data warehouse. Each database stores important information about insurance claims such as basic claim information, treatment information, disease information, and the filing review details. We provide details on each database in Table 2.1. From claims filed to HIRA in 2017, we extracted records that are manually reviewed. Also, we selected cases assigned to one of the following five 3-digit DRG codes for modeling and evaluation: B60(quadriplegia, paraplegia, and spondylopathy), B63(Parkinson disease, neurological neoplasm, hemiplegia, degenerative nervous system disorder), D64(disequilibrium, otitis media, upper respiratory infections), I07(simple spinal surgery, intervertebral disc removal), I68(non-surgical cervical and spinal conditions). Table 4.3 reports summary statistics of each DRG code group.

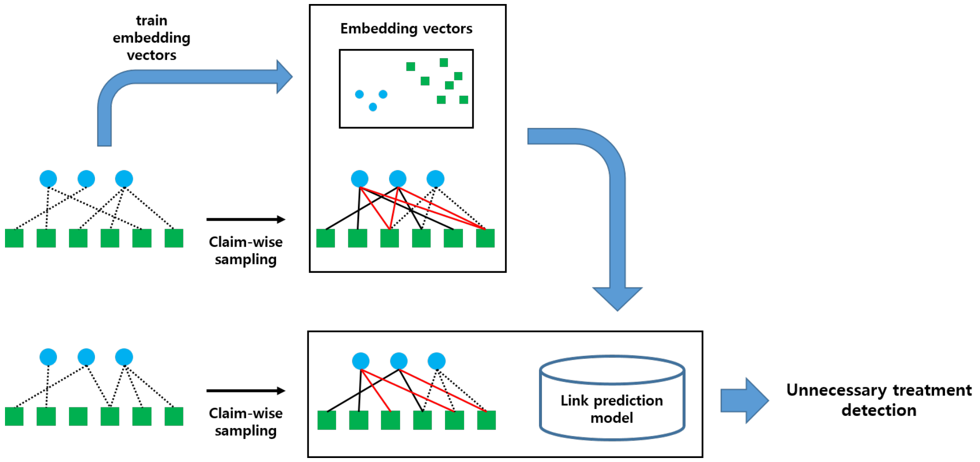


Figure 4.6: The process of unnecessary treatment detection by link prediction between the disease and the treatment

Table 4.3: Treatment data statistics

| Treatment type | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| Procedure | 804,306 | 1,562,496 | 1,951,333 | 4,504,776 | 10,586,623 |
| Prescription | 275,559 | 524,895 | 1,073,685 | 2,393,032 | 3,950,746 |
| Material | 17,384 | 16,947 | 32,712 | 509,183 | 40,753 |

1902 3-digit disease codes and 9765 treatment codes were included in the data set. Treatments are conventionally categorized into four different groups: basic treatments, procedure, prescription, and materials. Basic treatment refers to the group of treatments that may be prescribed anytime, regardless of types of diseases a patient is diagnosed with. For example, consultation, admission, nursing, and meals fall into this group. Since these types of treatments does not provide any meaningful information in relation to diseases, we discard them as we construct the disease-treatment network. Procedures are include treatments practitioners conduct on patients, such as X-ray examinations, MRI examinations, and surgical operations. Prescription refers to the detailed information about drugs practitioners prescribe such as, for example, the nonsteroidals anti-inflammatory drug. Finally, material is a group of treatments that require materials for recovery Orthosis is a good example of the material treatment. There are 5854, 2319, 1225 treatment codes in the procedure, prescription, material groups, respectively.

### 4.4.2 Experimental Settings

Our proposed method extracts disease-treatment relationship carefully by incorporating all the information available in the claim filing, instead of relying on simple co-occurrence. Multiple stages exploiting different information build up to the final disease-treatment network stage-by-stage. We begin by constructing networks separately for each DRG 3-digit code. Given a DRG 3-digit code, we extracted relevant RDRG codes, disease codes, procedure codes, prescription codes, and material codes that appear in our data. Then, we set each of these codes as individual nodes. RDRG code is of the finest granularity for DRG code in the KDRG code system. Then, for each RDRG code, we extracted the main disease codes and sub disease codes from all the claims with the matching the RDRG codes. Then, we connect the RDRG codes with the matching main disease codes. At the same time, we formed links between the main disease code nodes and the relevant sub-disease code nodes. Finally, we add treatment codes to the network and connect them with the associated disease codes as detailed in subsection 4.3.1. The diseases codes are grouped as detailed in subsection 4.4.1. Then the resulting network, by design, ensures that every treatment node is assigned to one of the three labels: procedure, prescription, or material. We present the possible types of disease-treatment pair edges in the resulting network in Table 4.4. We also provide the graphical snapshot of the network in Figure 4.7.

Table 4.4: Edges in the network and their type

| Edge | Type |
|---|---|
| RDRG - main disease | co-occurrences |
| main disease - sub diseases | co-occurrence |
| diseases - procedures | association |
| diseases - prescriptions | association |
| diseases - material | association |

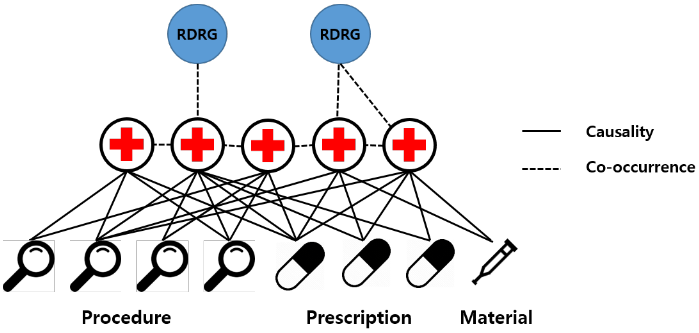


Figure 4.7: RDRG-disease-treatment network

In order to choose the best node embedding model to carry out the disease-treatment link prediction task, we trim the network by looking up the claims filed from January 2017 to September 2017 only. We split the edge sets into the training and the test set by the ratio of 7:3. The node embedding models we experimented with are as following: GF [2], HOPE [69], GraRep [9], DeepWalk [72], node2vec [29], metapath2vec [23], SDNE [92], LINE [86].

We report performances of various link prediction models employed, of which the respective unit link is defined to connect the disease and the corresponding procedure, the disease and the prescription, or the disease and the materials. We repeated the experiment for link prediction 10 times and report the average of the accuracy measures as the principal reporting metric for the overall performance. We choose the model with the highest average performance as the best embedding method for our network.

As for the overtreatment detection model as described in subsection 4.3.3, the training and the validation sets were built from the claims filed from January 2017 to September 2017. The test set comprises claims filed to HIRA from October 2017 to December 2017. We repeated the experiment 10 times and report the average accuracy as the performance metric.

We set hyper-parameters as follows. First, we fix the dimension of embedding vectors at 32 in every case. We tested various configurations and landed on the values reported. We set both the number of random walk per node and the length of each random walk at 32 for DeepWalk and node2vec methods. In case of node2vec, we set the two key hyper-parameters $p, q$, which, altogether, generate a biased random walk, at $p = 0.5, q = 2.0$. On the other hand, metapath2vec requires to define the meta-paths in order to generate random walks. We set the meta-paths to be either 'Treatment-Disease-Treatment', or 'Treatment-Disease-RDRG-Disease-Treatment'. The former meta-path implies that 'treatments caused by the same disease', while the latter, 'treatments for the same kind of patients'. We also fix the number of the random walk per node at 32. For SkipGram we set the window size for training at 6. For LINE, which considers information of neighboring nodes up to the 2nd order proximity, we set the negative ratio at 5. For GF, we fix parameter for the regularization term of the L2-norm loss at 0.00001. We used the 2-step transition probability matrix for GraRep method. The auto-encoder spart of the SDNE model takes the structure of $[n-256-32-256-n]$, where $n$ is the input dimension. Training batch size was set at 128 with the learning rate equal to 0.01. The parameters $\alpha, \beta$ in the loss function and $\nu$ in the regularization term is set to $\alpha = 0.1, \beta = 1.1, \nu = 0.3$, respectively We used pytorch [70], scikit-learn [71], scipy [91], numpy [67] packages to implement the aforementioned models.

## 4.5 Results

### 4.5.1 Network Construction

In this subsection, we compare the results between two distinct approaches to construct the disease-treatment network: (1) the simple co-occurrence-based approach, and; (2) the association-based approach. In previous subsections, we have claimed that simple co-occurrence per claim filings does not necessarily imply association. In this subsection, we will provide empirical justification for our arguments.

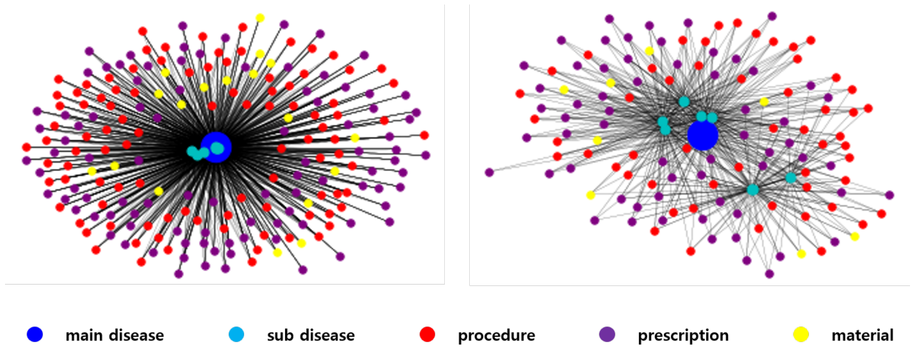


Figure 4.8: Networks constructed by co-occurrence and association relationship. (left): Co-occurrence (right): Association

Figure 4.8 represents the relationship between the main disease with code S12 (fracture of neck) and relevant sub-diseases from claim reports labeled with the DRG 3-digit code I07, whose linkage was determined by the two different approaches aforementioned. The left-hand side panel of Figure 4.8 shows the graphical representation of the relationships between the main disease S12 and the sub-diseases based on the simple co-occurrence, while the right-hand side panel shows that based on association. From now on, we refer to the left-hand side and the right-hand side network as the co-occurrence network and the association network, respectively.

In case of the co-occurrence network, every treatment which co-occurred with the main disease S12 under the DRG 3-digit code of I07 is linked not only to the main disease S12 but as well as to all the sub diseases which appeared with S12, as apparent on the left-side panel of Figure 4.8. On the contrary, treatment nodes from the association network clearly appear to be distributed more sparsely across the sub-diseases linked to the main disease S12, as shown on the right-side panel of Figure 4.8. We do not have the privilege of disclosing all the treatment nodes presented in Figure 4.8 due to personal information protection issues; yet, with permission from the appropriate authorities, we take an example from each network to provide an empirical justification for our argument. From the co-occurrence network, we have found that a link was formed between the main disease node noting a neck fracture with the treatment node for lumbar spine imaging. It is easy to see that there is no clear connection between the disease and the treatment mentioned. It is most likely due to the case in which a patient whose main disease of diagnose was the neck fracture, while lumbar spine imaging was prescribed for one of the sub-diseases not directly related or caused by the main disease.

On the contrary, for the association network, the number of direct linkages between the treatment nodes and the main disease node is far less than that of the co-occurrence network, while the different treatment types are dispersed throughout the range of the sub-diseases. Not surprisingly, the linkage between the fracture of neck and the lumbar spine imaging was missing from the association network.

### 4.5.2 Link Prediction between the Disease and the Treatment

In this subsection, we report the result of link prediction via node embedding. We trained a selection of node embedding models to learn vector representations for the nodes from the disease-treatment network and compared performance on solving the link prediction problem using the learned embedding vectors, the methodology of which is described in detail in subsection 4.3.2. Table 4.5, Table 4.6, and Table 4.7 reports each model's performance on the link prediction task using the disease-procedure, the disease-prescription, and the disease-material relations, respectively. In all cases, metapath2vec outperforms other models. This may be due to the characteristics peculiar to the network. Our network is constructed from a variety of information covering a rich range of different aspects of health insurance claims, hence strongly heterogeneous in nature. It is made of different types of nodes and edges. While other methods are devised for networks of homogeneous nature, the metapath2vec is designed to work well as heterogeneous network.

Based on the results from the link prediction test, we have selected the metapath2vec, of which the resulting embedding vectors are to be fed to the overtreatment detection model. The details on the overall mechanism of overtreatment model utilizing node embedding is elaborated in described in subsection 4.3.3.

Table 4.5: Link prediction results of disease-procedure

| Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| HOPE | 0.7337 | 0.7570 | 0.7768 | 0.7531 | 0.7662 |
| SDNE | 0.5769 | 0.5568 | 0.6162 | 0.3877 | 0.6357 |
| node2vec | 0.7426 | 0.7539 | 0.7795 | 0.7596 | 0.7720 |
| GraRep | 0.7338 | 0.7548 | 0.7798 | 0.7651 | 0.7747 |
| LINE | 0.7222 | 0.7386 | 0.7688 | 0.7609 | 0.7680 |
| GF | 0.1906 | 0.1970 | 0.1871 | 0.2037 | 0.1915 |
| DeepWalk | 0.7402 | 0.7534 | 0.7779 | 0.7621 | 0.6357 |
| metapath2vec | **0.8270** | **0.8357** | **0.8610** | **0.8534** | **0.8478** |

Table 4.6: Link prediction results of disease-prescription

| Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| HOPE | 0.7864 | 0.7927 | 0.7973 | 0.7808 | 0.8078 |
| SDNE | 0.6790 | 0.6532 | 0.6770 | 0.5320 | 0.7341 |
| node2vec | 0.7762 | 0.7890 | 0.7898 | 0.7731 | 0.8064 |
| GraRep | 0.7851 | 0.7907 | 0.8001 | 0.7847 | 0.8148 |
| LINE | 0.7794 | 0.7881 | 0.7896 | 0.7783 | 0.8063 |
| GF | 0.1701 | 0.1638 | 0.1732 | 0.1734 | 0.1672 |
| DeepWalk | 0.7721 | 0.7865 | 0.7879 | 0.7709 | 0.8067 |
| metapath2vec | **0.8155** | **0.8214** | **0.8297** | **0.8241** | **0.8487** |

### 4.5.3 Overtreatment Detection

Table 4.8 reports the performance test for the overtreatment detection. The term 'without embedding' refers to overtreatment detection models which utilizes the network resulting from the training set per se. The 'proposed method' refers to the overtreatment detection models which exploit node embedding methods to solve the link prediction problem, the mechanism of which is elaborated in detail in subsection 4.3.3. For most of the cases, the proposed model outperforms the 'without embedding' model. This may potentially imply that our proposed model performs better when some relational patterns are found only in the training set or in the test set, hence classifying freshly encountered treatments better.

Table 4.7: Link prediction results of disease-material

| Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| HOPE | 0.8328 | 0.8682 | 0.8961 | 0.7892 | 0.8679 |
| SDNE | 0.7122 | 0.7436 | 0.8563 | 0.5560 | 0.7340 |
| node2vec | 0.8511 | 0.8783 | 0.9110 | 0.7914 | 0.8881 |
| GraRep | 0.8363 | 0.8674 | 0.8991 | 0.7978 | 0.8756 |
| LINE | 0.8288 | 0.8607 | 0.8939 | 0.7878 | 0.8779 |
| GF | 0.7122 | 0.7436 | 0.8563 | 0.5560 | 0.7340 |
| DeepWalk | 0.8424 | 0.8753 | 0.9103 | 0.7992 | 0.8874 |
| metapath2vec | **0.9359** | **0.9467** | **0.9655** | **0.9038** | **0.9538** |

Table 4.8: Unecessary treatment detection by the network only and embedding vectors from the network

| Type of treatment | Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|---|
| Procedure | Without embedding | 0.9365 | 0.9346 | 0.9231 | 0.9387 | 0.8784 |
| | Proposed method | **0.9632** | **0.9667** | **0.9591** | **0.9768** | **0.9719** |
| Prescription | Without embedding | 0.8950 | 0.8844 | 0.9041 | 0.8983 | 0.8383 |
| | Proposed method | **0.9331** | **0.9010** | **0.9386** | **0.9238** | **0.9367** |
| Material | Without embedding | 0.8677 | **0.9230** | 0.8761 | 0.9168 | 0.8949 |
| | Proposed method | **0.9469** | 0.9177 | **0.8820** | **0.9547** | **0.8985** |

## 4.6 Summary

In the previous chapters, we proposed models for detecting abuse in medical treatments. These models, however, have yet to consider the relationship between diseases and treatments explicitly. Accounting for the disease-treatment relationship is important in a sense that, without doing so, detection models cannot properly process different drugs that have similar efficacy. There may be cases when different practitioners prescribe different drugs to a patient, where these drugs targets to alleviate the symptoms of the same disease. In order to process such cases appropriately, detection models need to be able to learn the intricate relationship between diseases and treatments.

This chapter presents a network-based approach through which the relationship between the diseases and treatments is considered during the abuse detection process. Our proposed model consists of three stages. During the first stage, a disease-treatment network is constructed based on information from claim filings. Since the association between diseases and treatments is not explicitly expressed, we infer the relationship by computing the relative risk (RR). Second stage involves selecting the best graph embedding method from several candidates available. We select the best method by comparing performances on link prediction. During the final stage, we solve a link prediction problem as the vehicle of overtreatment detection. If our link prediction model predicts links to be nonexistent for all of the diseases and treatments listed in a given claim, then the claim is classified as an overtreatment case.

We test the proposed model using the real-world claims data. Results show that the proposed method classify the treatment well which does not explicitly exist in the training network. The main contribution of this paper is that our model accounts for the disease-treatment relationship, which are not explicitly observed, during the process of overtreatment detection. Our model works well with practice patterns encountered the test phase only.

# Chapter 5

# Conclusion

## 5.1 Contribution

Abuse is a critical problem in the healthcare insurance industry. It refers to the medical service or the practice that is not consistent with the generally accepted sound fiscal practices. Reimbursing such cases cause waste of resources, eventually leading to the loss of the insurance company. Especially, abusive behaviors in national health insurance lead to social costs, which increase the premiums that the taxpayers have to pay. Therefore, detecting abuse behaviors and preventing compensation for them is a very important issue.

Currently, field professionals review the claims manually in order to screen out abuse cases. However, the astronomical increase in the number of claim filings is severely burdening the review process. Moreover, reviewing the claims require profound background knowledge and expertise, which makes the review process very costly. Adversities of such manual efforts calls for a more efficient review process. In response, past literature has employed various datamining techniques to automatically detect problematic claims or abusive providers. However, these studies do not utilize the treatment prescriptions, information of the finest granularity found in health insurance claims data. Existing studies relies on the claim-level or provider-level variables that are derived from the raw data, leading to relatively poor performance in detecting abusive claims.

The contributions of this dissertation is four-fold. Firstly, models we propose are based on medical treatment prescriptions, which is the lowest level of information available in the healthcare insurance claim. To our best knowledge, medical treatments have never been used in abuse detection. Using treatment prescriptions allows modelling abuse detection at various levels: treatment, claim, and provider-level. Secondly, we show that our finer-grained model outperforms models with higher level information. Thirdly, we propose a model which directly deals with seasonality, adding a realistic touch. Finally, we propose the abuse treatment detection model which account for the relationship between diseases and treatments, one of the most important information included in the medical treatment.

In chapter 2, we propose a scoring model based on which abusive providers are detected. Previous studies related to this topic rely primarily on provider-level variables. The coarse granularity of the mode leads to relatively poor performance. We propose the neural network-based scoring model that measures the degree of abuse for each provider. The model use treatments as input data. At the same time, we devise the evaluation metrics to quantify the efficiency of the review process. Experiment results show that the review process with the proposed model is more efficient than that with the previous model which uses the provider-level variables as input variables.

In chapter 3, we propose the method of detecting overtreatment and problematic claims under seasonality, which reflects more reality to the model. Several diseases are associated with seasonality. That is, in other words, the distribution claim is different from time to time. If the detection model does not consider this difference, its performance is not robust to the period in some departments. Instead of a single model for a department, we propose to a structure with multiple models built for several important DRG codes in the department. We test our proposed model using the real-world claim filings data, and results show that the proposed method is time-robust.

In chapter 4, we propose an overtreatment detection model accounting for the relationship between the disease and treatment. We discuss situations in which abuse detection may not work properly without the knowledge on the association relationship between disease and treatments. We propose an overtreatment detection approach method for detecting unnecessary treatment, which incorporating node embedding and link prediction methods. By solving the link prediction problem using the embedding vectors of nodes in the disease-treatment network, the model can infer pairs of disease and treatment unnecessarily reported in the insurance claims. We test our model using the real-world insurance claims data, and results show that our approach indeed works well with detecting claims with overtreatments. We additionally show that our model can be used in classifying the disease-treatment relationship.

## 5.2 Future Work

In this dissertation, we propose various abuse detection models based on the medical treatment prescription data. While our proposed models show satisfying results, there still is room for improvement. First of all, our current approach does not detect overtreatment on the claim-level. The underlying assumption here is that treatments listed in a claim are independent of one another. This may lack reality, since treatments can be prescribed to complement one another. If inference can be made on the level of individual treatments in a claim, more precise detection may be conducted.

Also, the proposed model in chapter 4 is for detecting totally unnecessary treatment, not for detecting necessary but overused treatment. In order to detect such treatment, we have to incorporate the proposed methods in this thesis. For example, train embedding vectors by graph embedding methods and train treatment classification model in chapter 2 or 3.

Finally, we can improve the performance by incorporating the data from external source. In chapter 4, we construct the disease-treatment network statistically using the claims data. However, it is unclear whether the constructed network has captured the true relationship. For example, suppose a practitioner prescribes several drugs to the patient. Some of the prescriptions may have been meant to complement each other. On the other hand, there may be the case in which the prescription includes a combination of drugs that causes side-effects when ingested together. The disease-treatment network we construct does not reflect such information. Due to the confidentiality contract, we could not utilize data from external sources as we conducted the study. However, it would help improve model performance if we could include external source data or knowledge graphs such as Drugbank[99], Twosides database [87], or SIDER [46] database.

# Bibliography

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, 2015. Software available from tensorflow.org.

[2] A. Ahmed, N. Shervashidze, S. Narayanamurthy, V. Josifovski, and A. J. Smola, *Distributed large-scale natural graph factorization*, in Proceedings of the 22nd International Conference on World Wide Web, WWW '13, New York, NY, USA, 2013, Association for Computing Machinery, p. 37–48.

[3] M. Alshahrani, M. A. Khan, O. Maddouri, A. R. Kinjo, N. Queralt-Rosinach, and R. Hoehndorf, *Neuro-symbolic representation learning on biological knowledge graphs*, Bioinformatics, 33 (2017), pp. 2723–2730.

[4] K. D. Aral, H. A. Güvenir, İ. Sabuncuoğlu, and A. R. Akar, *A prescription fraud detection model*, Computer Methods and Programs in

Biomedicine, 106 (2012), pp. 37 – 46.

[5] J. J. Baker, *Medicare payment system for hospital inpatients: diagnosis-related groups*, Journal of health care finance, 28 (2002), p. 1—13.

[6] A. Bayerstadler, L. van Dijk, and F. Winter, *Bayesian multinomial latent variable modeling for fraud and abuse detection in health insurance*, Insurance: Mathematics and Economics, 71 (2016), pp. 244 – 252.

[7] M. Belkin and P. Niyogi, *Laplacian eigenmaps and spectral techniques for embedding and clustering*, in Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, NIPS'01, Cambridge, MA, USA, 2001, MIT Press, p. 585–591.

[8] X. Cai, J. Gao, K. Y. Ngiam, B. C. Ooi, Y. Zhang, and X. Yuan, *Medical concept embedding with time-aware attention*, in Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, AAAI Press, 2018, p. 3984–3990.

[9] S. Cao, W. Lu, and Q. Xu, *Grarep: Learning graph representations with global structural information*, in Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM '15, New York, NY, USA, 2015, Association for Computing Machinery, p. 891–900.

[10] S. Cao, W. Lu, and Q. Xu, *Deep neural networks for learning graph representations*, 2016.

[11] Centers for Medicare & Medicaid Services and others, *Design and development of the diagnosis related group (DRG)*, 2018.

[12] P. S. Chan, M. R. Patel, L. W. Klein, R. J. Krone, G. J. Dehmer, K. Kennedy, B. K. Nallamothu, W. D. Weaver, F. A. Masoudi, J. S. Rumsfeld, R. G. Brindis, and J. A. Spertus, *Appropriateness of percutaneous coronary intervention*, JAMA, 306 (2011), pp. 53–61.

[13] H. Cho, B. Berger, and J. Peng, *Diffusion component analysis: Unraveling functional topology in biological networks*, in Research in Computational Molecular Biology, T. M. Przytycka, ed., Cham, 2015, Springer International Publishing, pp. 62–64.

[14] E. Choi, M. T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Tejedor-Sojo, and J. Sun, *Multi-layer representation learning for medical concepts*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1495–1504.

[15] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, *GRAM: Graph-based attention model for healthcare representation learning*, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, New York, NY, USA, 2017, Association for Computing Machinery, p. 787–795.

[16] Y. Choi, C. Y.-I. Chiu, and D. Sontag, *Learning low-dimensional representations of medical concepts*, AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science, 2016 (2016), p. 41—50.

[17] D. Clevert, T. Unterthiner, and S. Hochreiter, *Fast and accurate deep network learning by exponential linear units(ELUs)*, CoRR, abs/1511.07289 (2015).

[18] M. C. Cobanoglu, C. Liu, F. Hu, Z. N. Oltvai, and I. Bahar, *Predicting drug–target interactions using probabilistic matrix factorization*, Journal of Chemical Information and Modeling, 53 (2013), pp. 3399–3409. PMID: 24289468.

[19] G. Cybenko, *Approximation by superpositions of a sigmoidal function*, Mathematics of Control, Signals and Systems, 2 (1989), pp. 303–314.

[20] W. Dai, X. Liu, Y. Gao, L. Chen, J. Song, D. Chen, K. Gao, Y. Jiang, Y. Yang, J. Chen, et al., *Matrix factorization-based prediction of novel drug indications by integrating genomic space*, Computational and mathematical methods in medicine, 2015 (2015).

[21] J. Davis and M. Goadrich, *The relationship between precision-recall and roc curves*, in Proceedings of the 23rd International Conference on Machine Learning, ICML '06, New York, NY, USA, 2006, ACM, pp. 233–240.

[22] R. A. Derrig, *Insurance fraud*, Journal of Risk and Insurance, 69 (2002), pp. 271–287.

[23] Y. Dong, N. V. Chawla, and A. Swami, *Metapath2vec: Scalable representation learning for heterogeneous networks*, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,

KDD '17, New York, NY, USA, 2017, Association for Computing Machinery, p. 135–144.

[24] J. Duchi, E. Hazan, and Y. Singer, *Adaptive subgradient methods for online learning and stochastic optimization*, J. Mach. Learn. Res., 12 (2011), pp. 2121–2159.

[25] A. Ezzat, M. Wu, X.-L. Li, and C.-K. Kwoh, *Drug-target interaction prediction using ensemble learning and dimensionality reduction*, Methods, 129 (2017), pp. 81 – 88. Machine Learning Methods and Systems for Data-Driven Discovery in Biomedical Informatics.

[26] A. Ezzat, P. Zhao, M. Wu, X. Li, and C. Kwoh, *Drug-target interaction prediction with graph regularized matrix factorization*, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 14 (2017), pp. 646–656.

[27] Federal Bureau of Investigation, *Financial crimes report to the public: Fiscal year 2010-2011*, (Date accessed: 10 October 2018), 2012.

[28] D. N. Fisman, *Seasonality of infectious diseases*, Annual review of public health, 28 (2007), p. 127—143.

[29] A. Grover and J. Leskovec, *Node2vec: Scalable feature learning for networks*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 855–864.

[30] H. He, J. Wang, W. Graco, and S. Hawkins, *Application of neural networks to detection of medical fraud*, Expert Systems with Applications, 13

(1997), pp. 329 – 336. Selected Papers from the PACES/SPICIS'97 Conference.

[31] Health Insurance Review and Assessment Service, *2017 medical expense statistics*, (Date accessed: 10 October 2018), 2018.

[32] M. T. Hecker, D. C. Aron, N. P. Patel, M. K. Lehmann, and C. J. Donskey, *Unnecessary use of antimicrobials in hospitalized patients: Current patterns of misuse with an emphasis on the antianaerobic spectrum of activity*, Archives of Internal Medicine, 163 (2003), pp. 972–978.

[33] T. Hofmann and J. Buhmann, *Multidimensional scaling and data clustering*, in Proceedings of the 7th International Conference on Neural Information Processing Systems, NIPS'94, Cambridge, MA, USA, 1994, MIT Press, p. 459–466.

[34] K. Hornik, *Approximation capabilities of multilayer feedforward networks*, Neural Networks, 4 (1991), pp. 251 – 257.

[35] Z. Huang and N. Mamoulis, *Heterogeneous information network embedding for meta path based proximity*, 2017.

[36] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, *Fasttext.zip: Compressing text classification models*, arXiv preprint arXiv:1612.03651, (2016).

[37] L. Katz, *A new status index derived from sociometric analysis*, Psychometrika, 18 (1953), pp. 39–43.

[38] S. Kim, C. Jung, J. Yon, H. Park, H. Yang, H. Kang, D. Oh, K. Kwon, and S. Kim, *A review of the complexity adjustment in the korean diagnosis-related group (KDRG)*, Health Information Management Journal, 49 (2020), pp. 62–68. PMID: 30157672.

[39] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, CoRR, abs/1412.6980 (2014).

[40] T. N. Kipf and M. Welling, *Variational graph auto-encoders*, 2016.

[41] T. N. Kipf and M. Welling, *Semi-supervised classification with graph convolutional networks*, in 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, 2017.

[42] T. Kohonen, *Self-organization and associative memory*, vol. 8, Springer Science & Business Media, 2012.

[43] Korean Financial Supervisory Service, *In 2018, 800 billion won was caught for insurance fraud and 2.4 billion won in rewards*, (Date accessed: 01 December 2019), 2019.

[44] I. Kose, M. Gokturk, and K. Kilic, *An interactive machine-learning-based electronic fraud and abuse detection system in healthcare insurance*, Applied Soft Computing, 36 (2015), pp. 283 – 299.

[45] O. Kuchaiev, M. Rašajski, D. J. Higham, and N. Pržulj, *Geometric de-noising of protein-protein interaction networks*, PLoS computational biology, 5 (2009).

[46] M. Kuhn, I. Letunic, L. J. Jensen, and P. Bork, *The sider database of drugs and side effects*, Nucleic Acids Research, 44 (2015), pp. D1075–D1079.

[47] J. Lee, H. Shin, and S. Cho, *A medical treatment based scoring model to detect abusive institutions*, Journal of Biomedical Informatics, 107 (2020), p. 103423.

[48] B. E. Lehnert and R. L. Bree, *Analysis of appropriateness of outpatient CT and MRI referred from primary care clinics at an academic medical center: how critical is the need for improved decision support?*, Journal of the American College of Radiology : JACR, 7 (2010), p. 192—197.

[49] Y.-K. Lei, Z.-H. You, Z. Ji, L. Zhu, and D.-S. Huang, *Assessing and predicting protein interactions by combining manifold embedding with multiple information integration*, in BMC bioinformatics, vol. 13, Springer, 2012, p. S3.

[50] G. Li, J. Luo, Q. Xiao, C. Liang, P. Ding, and B. Cao, *Predicting microrna-disease associations using network topological similarity based on deepwalk*, IEEE Access, 5 (2017), pp. 24032–24039.

[51] X. Li, W. Chen, Y. Chen, X. Zhang, J. Gu, and M. Q. Zhang, *Network embedding-based representation learning for single cell RNA-seq data*, Nucleic Acids Research, 45 (2017), pp. e166–e166.

[52] C. Lin, C.-M. Lin, S.-T. Li, and S.-C. Kuo, *Intelligent physician segmentation and management based on kdd approach*, Expert Systems with Applications, 34 (2008), pp. 1963 – 1973.

[53] F.-M. Liou, Y.-C. Tang, and J.-Y. Chen, *Detecting hospital fraud and claim abuse through diabetic outpatient services*, Health Care Management Science, 11 (2008), pp. 353–358.

[54] Y. Luo, X. Zhao, J. Zhou, J. Yang, Y. Zhang, W. Kuang, J. Peng, L. Chen, and J. Zeng, *A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information*, Nature communications, 8 (2017), pp. 1–13.

[55] H. Lyu, T. Xu, D. Brotman, B. Mayer-Blackwell, M. Cooper, M. Daniel, E. Wick, V. Saini, S. Brownlee, and M. Makary, *Overtreatment in the united states*, PLoS One, 12 (2017).

[56] T. Ma, C. Xiao, J. Zhou, and F. Wang, *Drug similarity integration through attentive multi-view graph auto-encoders*, in Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, AAAI Press, 2018, p. 3477–3483.

[57] A. L. Maas, A. Y. Hannun, and A. Y. Ng, *Rectifier nonlinearities improve neural network acoustic models*, in Proc. icml, vol. 30, 2013, p. 3.

[58] M. E. Martinez, *The calendar of epidemics: Seasonal cycles of infectious diseases*, PLoS pathogens, 14 (2018), p. e1007327.

[59] J. M. McGinnis, L. Stuckhardt, R. Saunders, M. Smith, et al., *Best care at lower cost: the path to continuously learning health care in America*, National Academies Press, 2013.

[60] L.-A. McNutt, C. Wu, X. Xue, and J. P. Hafner, *Estimating the relative risk in cohort studies and clinical trials of common outcomes*, American Journal of Epidemiology, 157 (2003), pp. 940–943.

[61] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, 2013.

[62] T. Mikolov, W. tau Yih, and G. Zweig, *Linguistic regularities in continuous space word representations*, in HLT-NAACL, 2013, pp. 746–751.

[63] A. Mnih and G. E. Hinton, *A scalable hierarchical distributed language model*, in Advances in Neural Information Processing Systems 21, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds., Curran Associates, Inc., 2009, pp. 1081–1088.

[64] V. Nair and G. E. Hinton, *Rectified linear units improve restricted Boltzmann machines*, in Proceedings of the 27th International Conference on Machine Learning (ICML-10), J. Fürnkranz and T. Joachims, eds., Haifa, Israel, June 2010, Omnipress, pp. 807–814.

[65] C. Ngufor and J. Wojtusiak, *Unsupervised labeling of data for supervised learning and its application to medical claims prediction*, Computer Science, 14 (2013), p. 191.

[66] OECD, *OECD health statistics 2019*, 2019. `https://stats.oecd.org/Index.aspx?DataSetCode=SHA` (accessed on: 07 May 2020).

[67] T. E. Oliphant, *A guide to NumPy*, vol. 1, Trelgol Publishing USA, 2006.

[68] P. Ortega, C. Figueroa, and G. Ruz, *A medical claim fraud/abuse detection system based on data mining: A case study in chile*, vol. 6, 01 2006, pp. 224–231.

[69] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, *Asymmetric transitivity preserving graph embedding*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1105–1114.

[70] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, *Pytorch: An imperative style, high-performance deep learning library*, in Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.

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

[72] B. Perozzi, R. Al-Rfou, and S. Skiena, *Deepwalk: Online learning of social representations*, in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA, 2014, Association for Computing Machinery, p. 701–710.

[73] B. Perozzi, V. Kulkarni, H. Chen, and S. Skiena, *Don't walk, skip! online learning of multi-scale network embeddings*, 2016.

[74] L. Prechelt, *Early Stopping — But When?*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 53–67.

[75] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, *Struc2vec: Learning node representations from structural identity*, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, New York, NY, USA, 2017, Association for Computing Machinery, p. 385–394.

[76] S. T. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science, 290 (2000), pp. 2323–2326.

[77] W. J. Rudman, J. S. Eberhardt, W. Pierce, and S. Hart-Hester, *Healthcare fraud and abuse*, Perspectives in Health Information Management/AHIMA, American Health Information Management Association, 6 (2009).

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

[79] Y. Shan, D. Jeacocke, D. W. Murray, and A. Sutinen, *Mining medical specialist billing patterns for health service management*, in Proceedings of the 7th Australasian Data Mining Conference - Volume 87, AusDM '08, AUS, 2008, Australian Computer Society, Inc., p. 105–110.

[80] Y. Shan, D. W. Murray, and A. Sutinen, *Discovering inappropriate billings with local density based outlier detection method*, in Proceedings of the Eighth Australasian Data Mining Conference - Volume 101, AusDM '09, Darlinghurst, Australia, 2009, Australian Computer Society, Inc., pp. 93–98.

[81] Z. Shen, Y.-H. Zhang, K. Han, A. K. Nandi, B. Honig, and D.-S. Huang, *mirna-disease association prediction with collaborative matrix factorization*, Complexity, 2017 (2017).

[82] H. Shin, H. Park, J. Lee, and W. C. Jhee, *A scoring model to detect abusive billing patterns in health insurance claims*, Expert Systems with Applications, 39 (2012), pp. 7441 – 7450.

[83] L. Song, C. W. Cheong, K. Yin, W. K. Cheung, B. Fung, and J. Poon, *Medical concept embedding with multiple ontological representations*, in Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press, 2019, pp. 4613–4619.

[84] M. K. Sparrow, *Fraud Control in the health care industry: Assessing the state of the art*, US Department of Justice, Office of Justice Programs, National Institute of Justice Washington, DC, 1998.

[85] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, *Dropout: A simple way to prevent neural networks from overfitting*, Journal of Machine Learning Research, 15 (2014), pp. 1929–1958.

[86] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, *Line: Large-scale information network embedding*, in Proceedings of the 24th International

Conference on World Wide Web, WWW '15, Republic and Canton of Geneva, CHE, 2015, International World Wide Web Conferences Steering Committee, p. 1067–1077.

[87] N. P. Tatonetti, P. P. Ye, R. Daneshjou, and R. B. Altman, *Data-driven prediction of drug effects and interactions*, Science Translational Medicine, 4 (2012), pp. 125ra31–125ra31.

[88] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.

[89] T. Tieleman and G. Hinton, *Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude*, COURSERA: Neural networks for machine learning, 4 (2012), pp. 26–31.

[90] G. V. Trunk, *A problem of dimensionality: A simple example*, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1 (1979), pp. 306–307.

[91] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mul-

bregt, and S. . . Contributors, *Scipy 1.0: Fundamental algorithms for scientific computing in python*, Nature Methods, 17 (2020), pp. 261–272.

[92] D. Wang, P. Cui, and W. Zhu, *Structural deep network embedding*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1225–1234.

[93] M. Wang, M. Liu, J. Liu, S. Wang, G. Long, and B. Qian, *Safe medicine recommendation via medical knowledge graph embedding*, 2017.

[94] P. Wang, T. Hao, J. Yan, and L. Jin, *Large-scale extraction of drug–disease pairs from the medical literature*, Journal of the Association for Information Science and Technology, 68 (2017), pp. 2649–2661.

[95] S. Wang, H. Cho, C. Zhai, B. Berger, and J. Peng, *Exploiting ontology graph for predicting sparsely annotated gene function*, Bioinformatics, 31 (2015), pp. i357–i364.

[96] S. Wang, E. Huang, J. Cairns, J. Peng, L. Wang, and S. Sinha, *Identification of pathways associated with chemosensitivity through network embedding*, PLOS Computational Biology, 15 (2019), pp. 1–15.

[97] S. Wang, M. Qu, and J. Peng, *PROSNET: Integrating homology with molecular networks for protein function prediction*, 2017, pp. 27–38.

[98] S.-L. Wang, H.-T. Pai, M.-F. Wu, F. Wu, and C.-L. Li, *The evaluation of trustworthiness to identify health insurance fraud in dentistry*, Artificial Intelligence in Medicine, 75 (2017), pp. 40 – 50.

[99] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda, N. Assempour, I. Iynkkaran, Y. Liu, A. Maciejewski, N. Gale, A. Wilson, L. Chin, R. Cummings, D. Le, A. Pon, C. Knox, and M. Wilson, *Drugbank 5.0: a major update to the drugbank database for 2018*, Nucleic Acids Research, 46 (2017), pp. D1074–D1082.

[100] J. Xhang and K. Yu, *What's relative risk. a method of correcting the odds ratios in cohort studies of outcomes*, Journal of the American Medical Association, 280 (1998), pp. 1690–1691.

[101] Y. Xiang, J. Xu, Y. Si, Z. Li, L. Rasmy, Y. Zhou, F. Tiryaki, F. Li, Y. Zhang, Y. Wu, X. Jiang, W. J. Zheng, D. Zhi, C. Tao, and H. Xu, *Time-sensitive clinical concept embeddings learned from large electronic health records*, BMC medical informatics and decision making, 19 (2019), p. 58.

[102] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, *Prediction of drug–target interaction networks from the integration of chemical and genomic spaces*, Bioinformatics, 24 (2008), pp. i232–i240.

[103] W.-S. Yang and S.-Y. Hwang, *A process-mining framework for the detection of healthcare fraud and abuse*, Expert Systems with Applications, 31 (2006), pp. 56 – 68.

[104] Z.-H. You, Y.-K. Lei, J. Gui, D.-S. Huang, and X. Zhou, *Using manifold embedding for assessing and predicting protein interactions from high-throughput experimental data*, Bioinformatics, 26 (2010), pp. 2744–2751.

[105] A. Zell, *Simulation neuronaler netze*, vol. 1, Addison-Wesley Bonn, 1994.

[106] X. Zeng, N. Ding, A. Rodríguez-Patón, and Q. Zou, *Probability-based collaborative filtering model for predicting gene–disease associations*, BMC medical genomics, 10 (2017), p. 76.

[107] W. Zhang, Y. Chen, D. Li, and X. Yue, *Manifold regularized matrix factorization for drug-drug interaction prediction*, Journal of Biomedical Informatics, 88 (2018), pp. 90 – 97.

[108] X. Zheng, H. Ding, H. Mamitsuka, and S. Zhu, *Collaborative matrix factorization with multiple similarities for predicting drug-target interactions*, in Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13, New York, NY, USA, 2013, Association for Computing Machinery, p. 1025–1033.

[109] M. Zitnik, M. Agrawal, and J. Leskovec, *Modeling polypharmacy side effects with graph convolutional networks*, Bioinformatics, 34 (2018), pp. i457–i466.

[110] N. Zong, H. Kim, V. Ngo, and O. Harismendy, *Deep mining heterogeneous networks of biomedical linked data to predict novel drug–target associations*, Bioinformatics, 33 (2017), pp. 2337–2344.

[111] 조성준, 신훈식, 이제혁, and 안용대, *종합병원 심사 효율화를 위한 선정모형 개선 연구*, (2018).

[112] 조성준, 이제혁, 신훈식, and 김태욱, *전문심사 선정모형 개선방안 연구*, (2018).

# 국문초록

사람들의 기대수명이 증가함에 따라 삶의 질을 향상시키기 위해 보건의료에 소비하는 금액은 증가하고 있다. 그러나, 비싼 의료 서비스 비용은 필연적으로 개인과 가정에게 큰 재정적 부담을 주게된다. 이를 방지하기 위해, 많은 국가에서는 공공 의료 보험 시스템을 도입하여 사람들이 적절한 가격에 의료서비스를 받을 수 있도록 하고 있다. 일반적으로, 환자가 먼저 서비스를 받고 나서 일부만 지불하고 나면, 보험 회사가 사후에 해당 의료 기관에 잔여 금액을 상환을 하는 제도로 운영된다. 그러나 이러한 제도를 악용하여 환자의 질병을 조작하거나 과잉진료를 하는 등의 부당청구가 발생하기도 한다. 이러한 행위들은 의료 시스템에서 발생하는 주요 재정 손실의 이유 중 하나로, 이를 방지하기 위해, 보험회사에서는 의료 전문가를 고용하여 의학적 정당성여부를 일일히 검사한다. 그러나, 이러한 검토과정은 매우 비싸고 많은 시간이 소요된다. 이러한 검토과정을 효율적으로 하기 위해, 데이터마이닝 기법을 활용하여 문제가 있는 청구서나 청구 패턴이 비정상적인 의료 서비스 공급자를 탐지하는 연구가 있어왔다. 그러나, 이러한 연구들은 데이터로부터 청구서 단위나 공급자 단위의 변수를 유도하여 모델을 학습한 사례들로, 가장 낮은 단위의 데이터인 진료 내역 데이터를 활용하지 못했다.

이 논문에서는 청구서에서 가장 낮은 단위의 데이터인 진료 내역 데이터를 활용하여 부당청구를 탐지하는 방법론을 제안한다. 첬재, 비정상적인 청구 패턴을 갖는 의료 서비스 제공자를 탐지하는 방법론을 제안하였다. 이를 실제 데이터에 적용하였을 때, 기존의 공급자 단위의 변수를 사용한 방법보다 더 효율적인 심사가 이루어 짐을 확인하였다. 이 때, 효율성을 정량화하기 위한 평가 척도도 제안하였다. 둘째로, 청구서의 계절성이 존재하는 상황에서 과잉진료를 탐지하는 방법을 제안하였다. 이 때, 진료 과목단위로 모델을 운영하는 대신 질병군(DRG) 단위로 모델을 학습하고 평가하는 방법을 제안하

였다. 그리고 실제 데이터에 적용하였을 때, 제안한 방법이 기존 방법보다 계절성에 더 강건함을 확인하였다. 셋째로, 동일 환자에 대해서 의사간의 상이한 진료 패턴을 갖는 환경에서의 과잉진료 탐지 방법을 제안하였다. 이는 환자의 질병과 진료내역간의 관계를 네트워크 기반으로 모델링하는것을 기반으로 한다. 실험 결과 제안한 방법이 학습 데이터에서 나타나지 않는 진료 패턴에 대해서도 잘 분류함을 알 수 있었다. 그리고 이러한 연구들로부터 진료 내역을 활용하였을 때, 진료내역, 청구서, 의료 서비스 제공자 등 다양한 레벨에서의 부당 청구를 탐지할 수 있음을 확인하였다.

**주요어**: 부당청구 탐지, 건강보험, 진료내역, 딥러닝, 데이터마이닝

**학번**: 2014-30327

# 감사의 글

서울대학교 산업공학과의 모든 식구들께 감사드립니다.

공학박사학위논문

# Deep learning-based Abuse Detection in Healthcare Insurance with Medical Treatment Data

## 진료 내역 데이터를 활용한 딥러닝 기반의 건강보험 남용 탐지

2020 년 8 월

서울대학교 대학원

산업공학과

이 제 혁

# Deep learning-based Abuse Detection in Healthcare Insurance with Medical Treatment Data

## 진료 내역 데이터를 활용한 딥러닝 기반의 건강보험 남용 탐지

지도교수 조 성 준

이 논문을 공학박사 학위논문으로 제출함

2020 년 7 월

서울대학교 대학원

산업공학과

이 제 혁

이제혁의 공학박사 학위논문을 인준함

2020 년 7 월

위 원 장 이 재 욱 (인)

부위원장 조 성 준 (인)

위 원 이 경 식 (인)

위 원 강 필 성 (인)

위 원 고 태 훈 (인)

# Abstract

# Deep learning-based Abuse Detection in Healthcare Insurance with Medical Treatment Data

Jehyuk Lee
Department of Industrial Engineering
The Graduate School
Seoul National University

As global life expectancy increases, spending on healthcare grows in accordance in order to improve quality of life. However, due to expensive price of medical care, the bare cost of healthcare services would inevitably places great financial burden to individuals and households. In this light, many countries have devised and established their own public healthcare insurance systems to help people receive medical services at a lower price. Since reimbursements are made ex-post, unethical practices arise, exploiting the post-payment structure of the insurance system. The archetypes of such behavior are overdiagnosis, the act of manipulating patient's diseases, and overtreatments, prescribing unnecessary drugs for the patient. These abusive behaviors are considered as one of the main sources of financial loss incurred in the healthcare system. In order to detect and prevent abuse, the national healthcare insurance hires medical professionals to manually examine whether the claim filing is medically legitimate or not. However, the review process is, unquestionably, very costly and time-consuming. In order to address these limitations, data mining techniques have been employed to detect problematic claims or abusive providers showing an abnormal billing pattern. However, these cases only used coarsely grained information such as claim-level or provider-level data. This extracted information may lead to degradation of the model's performance.

In this thesis, we proposed abuse detection methods using the medical treatment data, which is the lowest level information of the healthcare insurance claim. Firstly, we propose a scoring model based on which abusive providers are detected and show that the review process with the proposed model is more efficient than that with the previous model which uses the provider-level variables as input variables. At the same time, we devise the evaluation metrics to quantify the efficiency of the review process. Secondly, we propose the method of detecting overtreatment under seasonality, which reflects more reality to the model. We propose a model embodying multiple structures specific to DRG codes selected as important for each given department. We show that the proposed method is more robust to the seasonality than the previous method. Thirdly, we propose an overtreatment detection model accounting for heterogeneous treatment between practitioners. We proposed a network-based approach through which the relationship between the diseases and treatments is considered during the overtreatment detection process. Experimental results show that the proposed method classify the treatment well which does not explicitly exist in the training set. From these works, we show that using treatment data allows modeling abuse detection at various levels: treatment, claim, and provider-level.

**Keywords**: Abuse detection, Healthcare insurance, Medical treatment, Deep learning, Data mining

**Student Number**: 2014-30327

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

As global life expectancy increases, spending on healthcare grows in accordance in order to improve quality of life. Figure 1.1 illustrates the annual expenditure on health per capita of several countries in OECD [66]. It can clearly be observed that the expenditure on healthcare is gradually increasing. In case of South Korea, healthcare expenditure per capita jumps to double between 2008 and 2018.

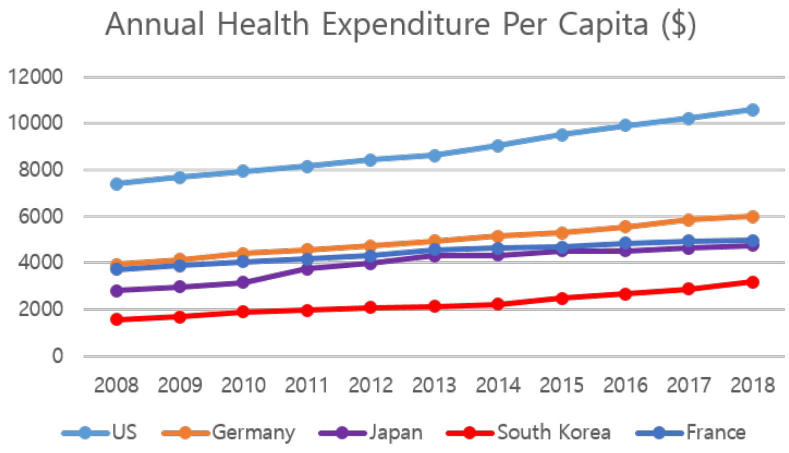


Figure 1.1: Annual expenditure on health per capita

However, medical care is quite expensive. Without a certain form of a compensation system, the bare cost of healthcare services would inevitably places great financial burden to individuals and households. In this light, many countries have devised and established their own public healthcare insurance systems to help people receive medical services at a lower price.

There exists a wild range of national healthcare insurance systems, and they differ by design from country to country. The first source of variability lies in the structure of the funding system. Canada and South Korea adopted the single-payer healthcare system, under which the government directly pays insurance fee by the means of general taxation. That is, in other words, citizens in these countries are legally obligated to pay taxes for the national health insurance. In France, compulsory contributions are made to make up the health insurance fund which are managed by non-profit organizations which are established solely for this purpose. In other countries, such as Germany or Belgium, a sickness fund is set up between employers and employees, and they make contributions to the fund. Under this system, funds are not from the government nor are they direct private payments.

On the other hand, the payment systems differ country by country. There is a variety of payment system structures including fee-for-service, bundled-payments, and global budgets. Fee-for-service refers to the payment system under which reimbursements are made for every treatment at a pre-determined unit price. This system is widely used because patients can receive quality care while providers can get reimbursed for their service. Bundled-payment, on the contrary, is a system which compensates medical expenses for the amount predetermined by the disease group to which the patient belongs to, instead of using the patient's medical history as a standard for compensation. The biggest strength of this system is that it is possible to suppress excessive treatment and increase transparency in medical expenses. Finally, global budget system estimates the total amount of medical expenses provided to the public and pays for the predetermined amount accordingly. Global budgets also has a tendency to reduce the likelihood of overdiagnosis or overtreatment. However, as compared to fee-for-service, the quality of medical care provided under bundled-payment or the global budget systems may be lower.

Given the trade-off between the quality of medical care and ethical practices, many countries run different systems simultaneously rather, than relying on a single system, in order to alleviate systematic shortcomings. France adopted all of the three systems, fee-for-service, global budget, and bundled payment system. Germany has established a modified version of global budgets and combined it with fee-for-service system. The health insurance system of South Korea, which will be the main focus of this study, takes the form of the fee-for-service system, which compensates the practitioners for their service. However, for seven disease groups, the reimbursement process takes the form of bundled-payment system, which compensates the practitioners by pre-determined payment, no matter how many treatments are provided to the patient. In other words, DRG-codes are incorporated in order to complement the limitations of the fee-for-service system with bundled payments-like apparatus. In order to broaden the scope of the system and the range of its application, the National Health Insurance Corporation of South Korea (NHIS) is implementing extensions to the bundled- payment system for patients who are not in pre-defined seven patient groups.

Since reimbursements are made ex post, unethical practices arise, exploiting the post-payment structure of the insurance system. The archetypes of such behavior are overdiagnosis, the act of manipulating patient's diseases, and overtreatments, prescribing unnecessary drugs for the patient. The loopholes in the system allow room for medical providers to prescribe excessive medical treatment or request non-existent medical treatment should they want to. Federal Bureau of Investigation (FBI) provides an extensive list of medical fraud cases by type, relevant health insurance information along with pertinent characteristic information, in the Financial Crime Report [27]. According to the report, the total fraudulent billing for health care programs amounts to be at least 3% of the total health expenditure which is estimated to be around \$2.4 trillion [27]. In addition, according to a report issued by the Korean Financial Supervisory Service, the amount of financial loss incurred by fraudulent activities was estimated to be about \$1.8 million in 2018, with further damages expected [43]. Such practices do not only increase the burden of medical expenses on the patients but also incur unnecessary social costs and expenditures. Some studies have reported that approximately 10% of medical spending is wasted due to these types of unethical practices ([22], [84]).

At this stage, let us clearly define fraud and abuse within the scope of healthcare insurance. These words appear ubiquitously in various situations, and it is difficult to disentangle the underlying meanings. Following the convention, We define fraud as a type of dishonest or intentional act which leads to unauthorized benefits for the person who commits the act or to someone else who is not entitled to the benefit [77]. On the other hand, we define abuse as a medical service or practice not consistent with the generally accepted sound fiscal practices [77]. Into the category of medical frauds fall the cases where medical service is documented and charged yet not really performed, or when a diagnosis on a patient is falsified in order to justify the unnecessary medical procedure. Abuse may lead to prescriptions that do not meet the medically stable criteria, or may result in incurring unnecessary costs by deliberately executing medically gratuitous treatments. Examples of abuse are overtreatment or improper billing practices.

Overtreatment, in particular, is considered as one of the main sources of financial loss incurred in the healthcare system. According to the report by Institute of Medicine, prescribing unnecessary services is the primary contributor to the loss incurred in the U.S. healthcare system to waste in US healthcare [59]. The report estimated that these behaviors account for approximately \$210 billion out of the \$750 billion loss in a year. Furthermore, a survey study, which collected survey the results from he Survey of overutilization of surveying 2106 physicians in the United States, about 20.6% of treatment is perceived as unnecessary [55]. Past literature has shown that such inappropriate or unnecessary treatments were especially conspicuous specialty care hospitals ([12], [32], [48]).

In order to detect and prevent abuse, the national healthcare insurance hires medical professionals to manually examine whether the claim filing is medically legitimate or not. However, the review process is, unquestionably, very costly and time-consuming. Moreover, there are not enough professionals to examine millions of claims. For example, in case of South Korea, there were only about 1,700 reviewers for 1.5 billion claims filed in 2016 [31]. Clearly, it is not possible to manually examine all the claims.

In response, insurance companies have resorted to an automated rule-based review system [79]. Although it can save much time and effort from reviewing all the claims manually, rule-based review system at the current level can only detect very simple abusive practices. Moreover, because this system is based on a set of predefined rules, it cannot cope with the new types of frauds and abuses rising over time.

In order to address these limitations, data mining techniques have been employed to detect abusive claims or providers showing an abnormal billing pattern ([4], [6], [30], [44], [52], [53], [65], [68], [79], [80], [82], [98], [103]). Based on these studies, insurance companies develop models detects abusive providers or problematic claims and examine relevant claims. However, these cases only used coarsely grained information such as claim-level or provider-level data. The lowest-level of information available from a claim, nonetheless, is the set of medical treatments, where patient's diseases and the set of corresponding medical activities are listed. A claim is a collection of several medical treatments, while abuse may be incurred as the result of a single or multiple medical treatments. Similarly, a provider can be represented by filed claims, while abuse may be incurred as the result of a single claim or multiple claims. So far, past literature has relied on the claim-level analysis or provider-level analysis, hence losing detailed information of each abuse in their detection models.

In this dissertation, we proposed an abuse detection methods in healthcare insurance using the medical treatment data, which is the lowest level information of the healthcare insurance claim. By using the lowest-level information, we show that it is possible to detect abuse from the healthcare insurance claims more precisely than the model with derived high-level variables. We also show that it is possible to detect abuse at various levels such as providers, claims, and treatments. First, we propose a scoring model based on which abusive providers are detected. We showed that the review process with proposed method is more efficient than with previous method, which is a scoring model with provider-level information. Second, we propose a detection model under the change of claim distribution, which reflects more reality condition. The proposed method is more robust to the change of distribution than the previous method. Third, we propose a detection model accounting for different prescription to the same patient, which reflects more reality condition. The proposed method understands the context of each entity by utilizing graph embedding method. The proposed model shows better performance than the model that does not include the context of each object. As can be seen here, we have proposed detection methods in situations similar to the real world.

This dissertation is organized as follows. In chapter 2, we proposed a neural network-based method of measuring the degree of abuse of medical service providers and selecting the abusive provider. Our model is, to our best knowledge, among the first to detect abuse in healthcare insurance using medical treatment data. In chapter 3, we propose overtreatment detection model which accounts for seasonality in claims by exploiting the concept of diagnosis-related groups, which was originally devised to classify patients. We showed that incorporating diagnosis-related groups during the claim review process helps detecting abuse better. In chapter 4, we propose an overtreatment detection model which extracts the relationship between the disease and the treatment by using graph embedding methods. Finally, we discuss the contributions and future work of this dissertation in chapter 5.

Table 1.1: Target abuse type, problem and proposed method covered in this dissertation

| **Chapter** | **Abuse unit** | **Problem** | **Proposed Methods** |
|---|---|---|---|
| Chapter 2 | Provider | Scoring the providers' degree of abuse | - Score the degree of abuse of provider with treatment data<br>- Used method: neural network with embedding layers |
| Chapter 3 | Treatment | Seasonality in the distribution of claims | - Operate the classification model by DRG code unit<br>- Used method: neural network with embedding layers |
| Chapter 4 | Treatment | Heterogeneous Treatment between Practitioners | - Modeling the disease-treatment relationship explicitly<br>- Used method: link prediction with graph embedding |

# Chapter 2

# Detection of Abusive Providers by department with Neural Network

## 2.1 Background

Abuse in healthcare refers to behaviors of providing unnecessary care to the patient. When an insurance company compensates for these unnecessary behaviors, it leads to the loss of the company. If this company is a national healthcare insurance company, it leads to an increase in premiums. In the case of countries with the single-payer healthcare system, such as South Korea, all taxpayers in the country will suffer from this loss. In other words, it can lead to a social loss in as sense that people cannot receive healthcare services at affordable prices. Due to this reason, abuse detection is an important task to solve for the healthcare insurance company, no matter if it is private or public.

In order to prevent the loss, they hire medical experts to detect these unnecessary behaviors. The problem is there are not enough experts to examine a bunch of claims. Moreover, to examine the healthcare claims, reviewers are required to know much more background knowledge than the other insurance. It means that it is more difficult to hire experts than other insurance. In order to tackle these difficulties, efforts were made to increase the efficiency of the review process. Instead of examining all claims carefully, reviewers select some problematic claims and manually review them. The objective here is set to reduce as much cost as possible by detecting as much abuse as possible with limited labor.

Now, the important issue now boils down to how these problematic claims should be selected. If a large proportion of the selected claims involves overtreatments, the reviewers can detect lots of abuse and reduce as much waste. If not, the effect of the examination would be insignificant. There have been studies that aim to detect these problematic claims using datamining techniques. These studies can be divided into two groups: detecting problematic claims and detecting abusive providers. The key assumption underlying the literature on 'detecting abusive providers' is that practitioners practice in a homogeneous pattern. This assumption would, in turn, lead to the conclusion that claims from abusive providers are more likely to include greater number of overtreatments. That is, in other words, if the reviewers can determine candidates for highly likely abusive providers and examine their claims as a priority, then there's a greater change to detect many more abuse claims in a shorter amount of time, hence being able to recover the loss induced by abuse. South Korea's HIRA screens through all the claim filings using their own scoring model to detect abusive provider candidates. The scoring model relies on datamining techniques, and the data the model learns is at the provider level.

However, previous studies do not use all of the detailed information residing in raw data. Past literature utilizes derived variables computed at the claim-level information or the provider-level. This can lead to poor performance of the model. Field experts from HIRA have expressed their aspiration to advance their existing model and suggest points of improvement. Their belief is that the primary reason of poor performance resides in the limitations in the input variables. Input variables currently in use are at the provider-level, hence incapable of accounting for different characteristics across providers. For example, there may be providers with a relatively small number of visits yet with large amounts of medical expenses, while some other providers have a relatively large number of visits with small amounts of medical expenses. Patient visits and medical expenses may vary according to the size of the provider. Failure to account for provider-wise variations may lead to degradation of the model's performance. Moreover, different diseases may be associated with different forms of abuse, yet provider-level variables cannot account for disease-wise variations either.

In this chapter, we address these issues and propose a model that scores the degree of medical abuse by provider using medical treatment data. The proposed method consists of two steps: training a neural network which scores the degree of abuse from each medical treatment, and then calculating the abuse score of each treatment by multiplying the neural network result by the claimed amount. Finally, abuse scores of the treatments are aggregated to the provider level. We define the resulting score to be the abuse score of the subject provider. We test the proposed model using in-patient claim data from six different departments in the year of 2016. Experiment results show that the proposed model is more efficient than the existing model which uses only provider-level variables. In addition, we show that the proposed model scores providers well as compared to the previous model.

The rest of the chapter is organized as follows. In section 2.2, we review the past literature on data mining methods for abnormality detection in healthcare insurance. Section 2.3 provides detailed descriptions of the proposed model. In section

2.4, we elaborate on experiment settings. We also describe the devised evaluation measures in this section. Section 2.5 reports the experiment results. Finally, section 2.6 concludes the paper.

## 2.2 Literature Review

### 2.2.1 Abnormality Detection in Healthcare Insurance with Datamining Technique

There are many studies on detecting fraud or abuse in the health insurance industry. In this subsection, we briefly survey through two major branches of health insurance abnormality detection: detecting abnormal providers, detecting abnormal claims.

**Detecting abnormal providers**

First, we briefly review several studies related to detecting abnormal providers. Here, the term 'provider' means medical service provider which provides medical service to patients such as medical institutions, general practitioners. We define abnormal providers as the providers that have different billing patterns to others. Most studies that aim to detect these providers suggest models with provider-level variables. In most cases, these variables are extracted from the raw data.

He et al. [30] applied the multi-layer perceptron (MLP) to detect abnormal General Practitioners (GPs) with sampled profile data of practicing GPs. They used 28 GP-level features that are selected by consultants. Also, the profiles were labeled on a 1-4 scale. They trained a multilayer perceptron that with this data. Also, they utilized the self-organized map (SOM) [42] with the MLP to classify the GP practice profiles.

Shan et al. [79] used the association rule mining method to make rules for detecting abnormal providers. These rules include both positive rules and negative rules. They applied the method in real claim data from Medicare Australia's Enterprise Warehouse. As a result, they extracted 215 rules and evaluated qualitatively and quantitatively. Users are willing to use this method because they can interpret the abuse though they can detect only simple abuses with these rules.

Shan et al. [80] detected abnormal providers by utilizing the local outlier factor (LOF) method which is a kind of unsupervised learning approach. They applied the method in the Australian optometrist dataset using 12 provider-level variables. They found the proposed approach outperforms domain-knowledge based methods. It means even if data is not fully labeled, the unsupervised method may be a good method to detect providers with abnormal billing patterns.

Liou et al. [53] conducted a study of detecting abnormal providers with extracted cost-related variables such as average drug cost, average diagnosis fee, or average medical expenditure per day. They trained three supervised learning models with the claim data from Taiwan's National Health Insurance using these variables. The three models were logistic regression, neural network, and classification tree. They found that the proposed model classifies abnormal providers from all providers well.

Lin et al. [52] suggested a knowledge discovery in database (KDD) approach based method that aims to detect abnormal GPs. The proposed method includes these processes. First, extract GPs' profiles from the claim databases. Here, the profile means the provider-level information such as the amount of fee, amount of prescription days, or average drug fee per case. From these variables, segment the providers using clustering methods such as SOM or PCA. Then, they described the billing patterns of each segment and provided the detailed managerial guidance that is from domain experts. They selected abnormal segments based on this guidance. Finally, the providers in such segments are considered as abnormal providers. They applied this method to the claim data from the National Health Insurance of Taiwan. The result was promising in that the model detects the abnormal providers efficiently.

Shin et al. [82] devised a scoring model that scores the provider's degree of abuse. Also, they claimed that the score from their model can be used to detect the abusive providers. The scoring model is includes following steps. First, calculate the degree of anomaly (DA) for each variable which means the deviation from the average value. Then, define the composite degree of anomaly (CDA) as a weighted average of DA and calculate CDA for each provider. The provider's CDA value is considered as the provider's degree of abuse. After the CDA value for each provider is calculated, derive the grade for degree of anomaly (GDA) by segment the CDA into several groups. In order to use these scores in detecting abusive providers, train a decision tree model using provider's profiles as input variables, and GDA value as a target variable. They applied this method to the outpatient claim data from HIRA in South Korea. They found that the proposed model is able to detect abusive providers well and easy to update.

These studies are about detecting abnormal providers using each provider's profile information. In other words, these models only use each provider's information, not the provider-provider relationship. A study conducted by Wang et al. [98] is about detecting abusive providers using the relationship. They constructed a social network of patients and providers from patients' visit sequences. For example, suppose a patient visits provider A. If the patient shows no improvement, he may visit different provider B. If then, make a directed edge from provider A to provider

B. It means if a node has high out-degree, the provider corresponding to the node can be considered suspicious. After the network is constructed, calculate the trustworthiness score for each node. Then, select suspicious providers and consider them as abnormal providers. They applied this method to both simulated and real-world claim data from National Health Insurance of Taiwan. They found that their method is effective in identifying abnormal providers. Also, they claimed that reviewers can detect abnormal providers effectively if the proposed method is used with traditional methods.

**Detecting abnormal claims**

We can define abnormal claims as the claims that have different billing patterns to others or including overtreatment. Most papers that aims to detect these claims suggest models with claim-level variables. The variables are also extracted from the raw data. Yang and Hwang [103] used the process mining framework to detect abnormal claims. They define a term clinical pathway, which means frequent clinical patterns from clinical instances. If an instance deviated from the pathway, it is considered as an abnormal claim. They applied the proposed method in claim data from NHI program of Taiwan. Their experiment shows that the proposed method is more efficient than manually constructed detection models.

Ortega et al. [68] suggest a framework that detects abnormalities using the neural network. This model is not aimed to detect abnormal claims only. It aims to detect 'abnormalities'. First, they define four import entities that play important roles in healthcare insurance: medical claims, affiliates, medical professionals, and employers. Then, train neural networks for each entity that detects abnormalities. Also, the classification model and result from one entity give feedback to other models to improve the model performance. They applied the proposed framework in real claim data from private pre-paid health insurance plans(ISAPRE) in Chile. They found that the proposed model shows better performance. Also, the model shows good performance even if a new input data has quite different patterns to previous data.

Aral et al. [4] supposed a model that calculates the fraudulent risk of a claim with cross-feature analysis. The fraudulent risk is calculated from incidence matrices that are derived from a correlated variable pair. Risk metrics from both categorical features and ordinal features are calculated from the incidence matrices. They applied the proposed method to real claim data from Turkey. It shows that their approach is capable of detecting abnormalities. Moreover, another important feature of the model is the fact that it can be used in online because the inference time of this model is very short.

A framework that Bayerstadler et al. [6] devised is quite different. They try to model the claim with Bayesian multinomial latent variable. They assumed every claim is in one of following categories: 'Unperformed services'(UP), 'Unjustified services'(UJ), 'Other billing issues'(BI), and 'No irregularities'(NI). Then, a claim $i$ follows the multinomial distribution with several parameters. In order to estimate these parameters, they used the multinomial logit model and Markov Chain Monte Carlo (MCMC) sampling. They confirmed that the performance of their model showed better performance than other benchmark models.

One of the weaknesses of previous models is that they are not proper models to detect evolving frauds or abuses. In order to detect these changing abnormalities, the model needs to be re-training. However, it is difficult to know the right time to retrain, because medical claim data is different from stream data. Ngufor and

Wojtusiak [65] suggested a change point detection model with the concept drift method to solve this problem. Also, they suggested an abnormal claim detection method after detecting these change points. They applied the proposed method to simulated data and real claim data from INOVA Health System of Northern Virginia.

The approach of Kose et al. [44] suggested is quite different from previous models. They asserted that fraudulent behavior should be considered as provider-claim pair, not the claim or provider itself. They claimed that frauds are from the behaviors of multiple actor types (providers) and multiple commodities(claims). Also, they utilized the interactive machine learning approach in order to make the model adapt to changing fraud types. They found the proposed method is capable of detecting abnormalities well.

### 2.2.2 Feed-Forward Neural Network

The feed-forward neural network is a type of an artificial neural network, in which the nodes in the model do not form a cycle [105]. In other words, the information only moves from input nodes to output nodes through hidden nodes without any backward moves. It is the simplest form of the artificial neural network. If there is only a single layer of output nodes, it is called a single-layer perceptron network. If the network consists of several layers, then it is called a multilayer perceptron (MLP) network.

The main goal of the feed-forward network is to approximate a function. According to the universal approximation theorem, a feed-forward neural network comprising a single hidden layer with an activation function and a linear output layer can approximate continuous functions on the compact subset of $\mathbf{R^n}$ ([19], [34]). That is,

Table 2.1: Previous studies about the abnormal detection in healthcare insurance

| Type of abnormality | Authors | Data mining approach | Method |
|---|---|---|---|
| Provider | He et al. [30] | Supervised | Neural network |
| | Lin et al. [52] | Unsupervised | PCA, SOM |
| | Liou et al. [53] | Supervised | Classification tree, logistic regression, neural network |
| | Shan et al. [79] | Unsupervised | Association rules |
| | Shan et al. [80] | Unsupervised | Local density based outlier detection |
| | Shin et al. [82] | Supervised | Distance based method for univariate variable, decision tree |
| | Wang et al. [98] | Unsupervised | Network analysis |
| Claim | Yang and Hwang [103] | Supervised | Process mining, association rules |
| | Ortega et al. [68] | Supervised | Neural network |
| | Aral et al. [4] | Hybrid | Distance based correlation and risk matrices |
| | Ngufor and Wojtusiak [65] | Hybrid | Change point detection, unsupervised data labeling, classification model |
| | Bayerstadler et al. [6] | Supervised | Latent variable modeling, MCMC |
| Behavior (Providers-claims) | Kose et al. [44] | Interactive | Analytic hierarchical processing(AHP), EM algorithm, data visualization |

in other words, a large MLP may represent any function given proper parameters. However, it does not guarantee that the training algorithm will be able to learn that function for sure.

In many cases, the back-propagation algorithm is used to train a neural network [78]. When the input value $\boldsymbol{x}$ generates an output value $\hat{y}$, the scalar error $E(\boldsymbol{\theta})$ is calculated, with $\boldsymbol{\theta}$ representing the set of parameters in the model. The back-propagation algorithm allows moving this information from the output layer to the input layer while computing the gradient. The network parameters are updated according to these gradients by $\Delta\boldsymbol{\theta} = -\alpha \cdot (\delta E(\theta)/\delta\boldsymbol{\theta})$, in order to minimize the error function.

Consider a $m$-layer feed-forward neural network which is fully connected. The input dimension is $n$, and the output dimension is 1. Let us define some notations as follows.

- $w_{ij}^k$: weight for perceptron $j$ in the $k$-th hidden layer for the incoming node $i$ in the $(k-1)$-th hidden layer
- $b_i^k$: bias of perceptron $i$ in the $k$-th hidden layer
- $h_i^k$: the product sum plus the bias of perceptron $i$ in the $k$-th hidden layer
- $g_h$: activation function of the hidden layers
- $g_o$: activation function of the output layers
- $o_i^k$: the output of the node $i$ in the $k$-th hidden layer
- $r_k$: number of the nodes in the $k$-th hidden layer
- $\boldsymbol{w_i^k}$: weight vector of perceptron $i$ in the $k$-th layer. $\boldsymbol{w_i^k} = \{w_{1i}^k, ..., w_{r_k i}^k\}$
- $\boldsymbol{o^k}$: output vector of $k$-th layer. $\boldsymbol{o_k} = \{o_1^k, ..., o_{r_k i}\}$

Then, the output of the neural network can be expressed as follows

$$\hat{y} = g_o(\boldsymbol{w_i^m} \cdot \boldsymbol{o^{m-1}} + b_1^m)$$

Where $h_i^k = \boldsymbol{w_i^k} \cdot \boldsymbol{o^{k-1}} + b_i^k, o_i^k = g_h(h_i^k)$, for $i = 1, 2, ..., r_k$

## 2.3 Proposed Method

This section presents a scoring model that measures provider's degree of abuse by using medical treatments. The model should give higher score to more abusive providers if the model is well trained. Then, the review process might be efficient if the reviewers only review claims from providers with high score.

At this point, we clearly define the degree of abuse. Once a provider submits a claim to the insurance company, the reviewers examine all the treatments appearing in the claim. Then, they determine whether each treatment is abused or not. If a treatment is adjudged as abuse, the amount of abuse is determined in following way: if the treatment is considered to be totally unnecessary to the patient, the abused amount equals the amount claimed; if the treatment is considered to be necessary yet excessive, then the abused amount is less than the claimed amount. The insurance company reimburse the providers for the total claimed amount, excluding the abused amount. In this paper, we define the degree of abuse of a provider as the total abused amount from whole claims that is submitted by the provider.

The proposed method consists of two steps. First, we train a model that calculates the likelihood of abuse for each medical treatment. The model is a kind of neural network that classifies whether the treatment is normal or abuse. Upon the completion of calculating the likelihood for each treatment in the test set, the result form the neural network is multiplied by the claimed amount the resulting measure of which we define as the abuse score of the subject treatment. Then, aggregate the abuse score for each treatment to the provider-level by combining scores if the treatments came from the same provider. We define the result as the abuse score of the provider. Figure 2.1 summarizes the whole framework of the provider's degree of abuse.

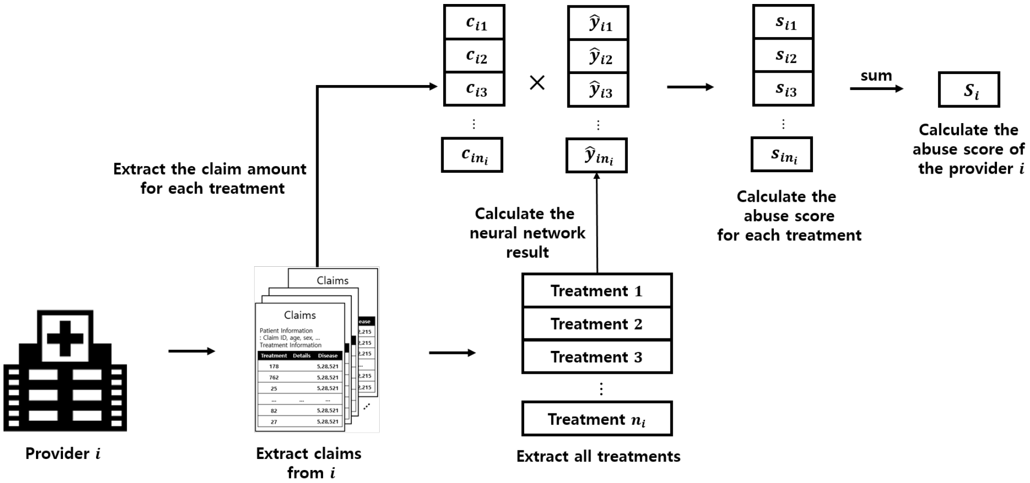


Figure 2.1: The process of scoring a provider's degree of abuse

### 2.3.1 Calculating the Likelihood of Abuse for each Treatment with Deep Neural Network

The proposed model employs a deep neural network to calculate the likelihood of abuse for each treatment. The model uses the documented information regarding to each treatment as input variables. The input variables include patient-related information, medical treatment-related information. The patient-related information includes age, gender, diseases, as well as the medical treatment-related information includes the type of operation, the unit price of medicine, or the number of medication days. The model structure is illustrated in Figure 2.2.

As illustrated in Figure 2.2, the proposed model uses both numerical variables and categorical variables. One of the most common-approaches to deal with cat-

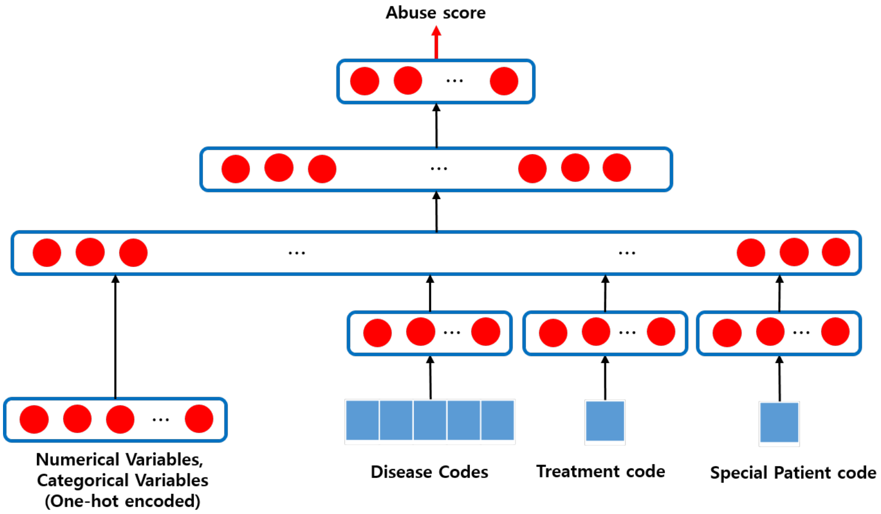


Figure 2.2: Structure of treatment scoring model

egorical variables is one-hot encoding. However, this method is undesirable if the categorical variables are of high cardinality. The data dimension will be exploded if we convert the categorical variables to numerical vectors using one-hot encoding method. Instead, we hire an embedding function to convert those variables. Our proposed model trains the embedding function as part of the training phase of the entire network. The classification error is back-propagated to embedding layers as well as hidden layers. We illustrate the training phase mathematically as follows. Suppose that we want to represent a category variable with cardinality $V$ as a $d$-dimensional vector, which is $d \ll V$. The embedding vector can be calculated as follows.

$$\boldsymbol{h} = f(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{W}$$

Here, the embedding matrix $\boldsymbol{W}$ is also trained with the neural network as it minimizes the total error function during the training phase. In this case, the total loss takes the form of a binary cross entropy function, defined as follows.

$$\mathbf{L} = -\sum_{i=1}^{N}(y_i \log \hat{y_i} + (1 - y_i) \log(1 - \hat{y_i})$$

The classification error is back-propagated through the hidden layers and the embedding layers, and the parameters for both of the layers are updated as error is minimized.

In our model, we also account for a special case: the multi-valued categorical variable. In this study, the patients' disease information variable has such characteristics. When a patient visits the medical provider, his/her medical record for the visit is mostly likely to be associated with more than one disease. In this case, the disease variable has multiple values.

If the claim data includes the association relationship between disease and treatment, it would be easy to determine whether the treatment is appropriate to the patient. Unfortunately, most of the claims do not include this relationship information. Moreover, it is difficult to disentangle medical activities one by one and determine its relationship to the corresponding disease explicitly, because it may be the case that some activities are prescribed under the consideration of the potential interaction of multiple diseases. Instead, the claim includes the all diseases of the patient and all treatments details. In order to utilize treatment data in modeling, all diseases in the claim should be matched with every treatment in the claim.

Due to the lack of relationship information between disease and treatment, we average the embedding vector by disease category to represent the diseases as nu-meric vectors. The method for calculating the embedding vector for each disease code and other high-cardinality category variables are illustrated in Figure 2.3.

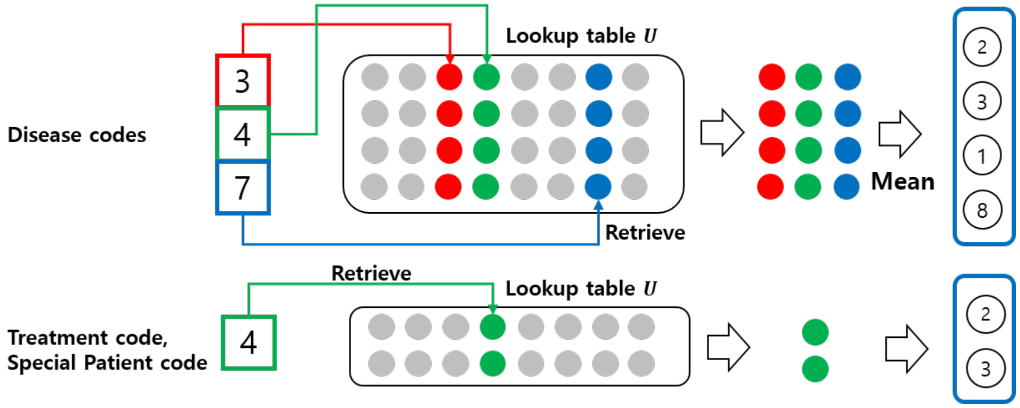


Figure 2.3: Embedding method of categorical variables with high-cardinality

### 2.3.2 Calculating the Abuse Score of the Provider

In this subsection, we describe the process of computing abuse score for each provider based on the calculated result from the neural network, which we described in subsection 2.3.1. In this subsection, we describe the process for computing the abuse score for each provider based on the results from the neural network. Suppose there are $N$ providers and with $m_1, m_2, \ldots, m_N$ claims, and $n_1, n_2, \ldots, n_N$ treatments. The amount claimed for $j$-th medical treatment by provider $i$ is represented by $c_{ij}$. The abuse likelihood of the treatment calculated by the medical treatment scoring model is represented by $\hat{y}_{ij}$. Now, the abuse score of the provider $i$ is computed in two steps: (1) the abuse score is determined for each treatment ($s_{ij}$), and then (2) the abuse scores are aggregated across treatments if they came from the same provider ($S_i$). Above two steps are summarized below.

- Calculate the abuse score of each treatment ($s_{ij}$)

$$s_{ij} = c_{ij}\hat{y}_{ij}, j = 1, 2, \ldots, n_i, i = 1, 2, \ldots, N$$

- Calculate the abuse score of each provider ($S_i$)

$$S_i = \sum_{j=1}^{n_i} s_{ij}, j = 1, 2, \ldots, n_i, i = 1, 2, \ldots, N$$

$S_i$ represents the abuse score of provider $i$, which measures the degree of abuse by the provider $i$. In order to maximize the efficiency in the reviewing process, we include the amount claimed when calculating the abuse score. Suppose there are two providers with the same number of treatments with the same abuse likelihood. Now, suppose the claimed amount for each treatment of one provider is larger than that of the other. Then, it is likely that the social cost incurred by the abuse of the former provider is larger than the latter. If reviewers can detect such abuse, the social benefit from the former is larger than the latter. This means that the reviewers can examine efficiently in a sense that they can detect abuse cases with greater social cost with smaller amount of input labor.

## 2.4 Experiments

We apply the proposed method to real-world claims data, which were submitted to HIRA in 2016. We compare the performance of the proposed model against the previous model employed by HIRA, which utilizes provider-level variables. In subsection 2.4.1, we provide the detailed descriptions of the data. Training details can be found in subsection 2.4.2. In subsection 2.4.3 and 2.4.4, we describe the evaluation measures devised for proper performance comparison among different models.

### 2.4.1 Data Description

In this dissertation, we used healthcare insurance claims submitted to the NHIS for experiments. Before we introduce our work, we provide detailed description of the filing and review process of health insurance claims.

The majority of South Korea citizens are covered by a uniform health insurance policy administered by NHIS. When a patient receives medical care from a medical service provider, the provider submits the claim for reimbursement of the amount determined by the fee-for-service policy. Then, the patient only pays for the remaining amount. Although the government strictly regulates the reimbursement process, abuse cannot be perfectly prevented, which eventually causes waste of the healthcare budget. The Health Insurance Review and Assessment (HIRA) is an institution dedicated for the detection of such abuses by investigating medical claims and auditing medical institutions. Once a medical provider submits a claim to HIRA, the reviewers examine the claim and determine whether the claim is suspicious of abuse or fraud. Then, HIRA submits the examination results to the NHIS where the reimbursements are made in accordance with the results to the subject provider. The reimbursement process is represented in Figure 2.4.

HIRA takes several steps when examining claims. When a claim is filed to HIRA, an automatic system initially checks whether there is an error in the basic information of the claim. This process is referred to as the automatic checkup process. Then, the claim goes under the process called an electronic review. In this step, a model

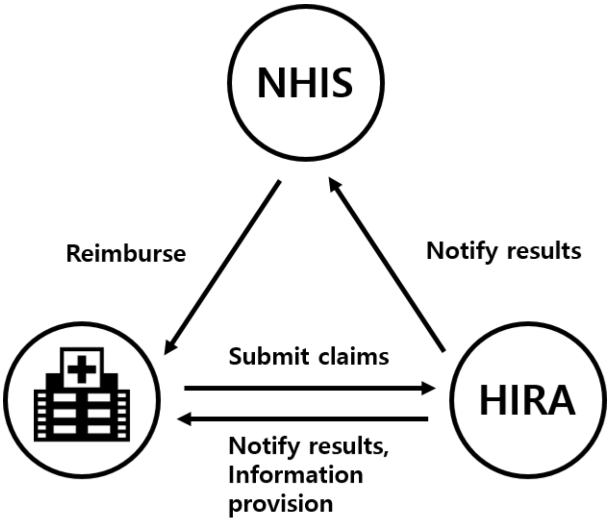
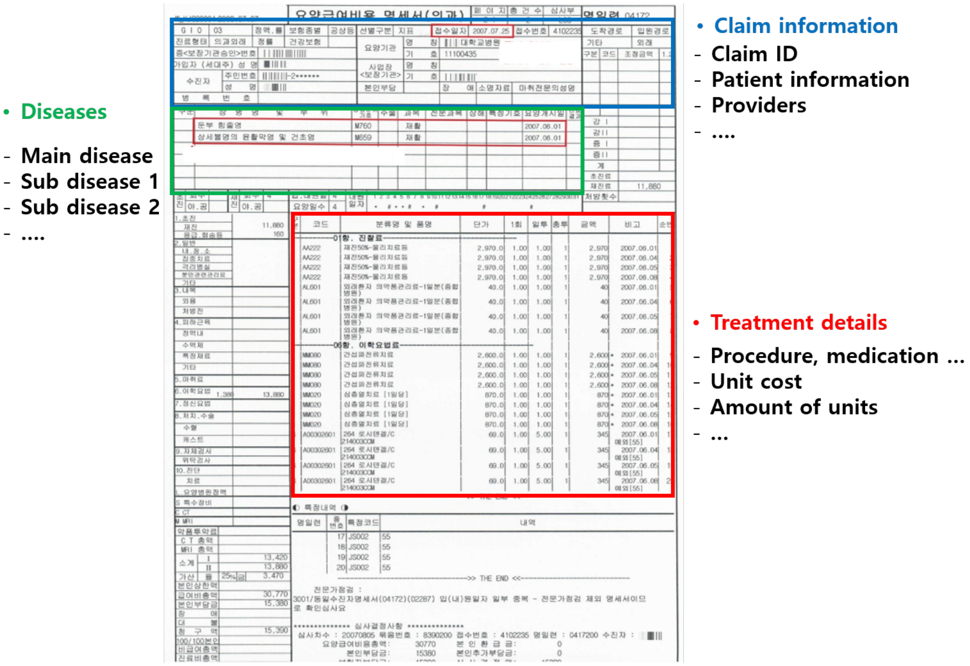


Figure 2.4: The reimbursement process of NHIS

detects whether the claim is abuse claim or not in seven steps. The model is based on the reviewer's experience. After this process, it goes through one of two processes. If it is considered as a normal claim, the result of the examination is sent to the NHIS. Then, NHIS reimburses the provider. If the claim is considered to be suspicious, it goes through the manual review. In this process, the reviewers manually examine the claim one by one. Manual review involves two kinds of examination:. the regular examination, and the irregular examination. In the regular examination process, reviewers select several abusive providers and manually review all the claims from them. Here, the abusive providers are selected by the datamining model of HIRA's own device, which uses provider-level information as input variables. In the irregular examination, reviewers select several important and complex claims and review them precisely. If the claim is much more complex than the others, it goes through


• Diseases
- Main disease
- Sub disease 1
- Sub disease 2
- ....
• Claim information
- Claim ID
- Patient information
- Providers
- ....
• Treatment details
- Procedure, medication ...
- Unit cost
- Amount of units
- ...

요양급여비용 명세서(의과)

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 페이지 | 총건수 | 심사부 | 명일련 04172 | | | | | | |
| G I 0 | 03 | 정액.률 | 보험종별 | 공상등 | 선별구분 | 지표 | 접수일자 | 2007.07.25 | 접수번호 4102235 |
| 진료형태 | 의과외래 | 정률 | 건강보험 | | 요양기관 | 명칭 | [redacted] 대학교병원 [redacted] | | |
| 증<보장기관승인>번호 | [redacted] | | | | | 기호 | 11100435 | | |
| 가입자 (세대주) 성명 | [redacted] | | | | 사업장<보장기관> | 명칭 | | | |
| 수진자 | 주민번호 | [redacted]-2****** | | | | 기호 | [redacted] | | |
| | 성명 | [redacted] | | | 본인부담 | | 장애 | 소명자료 | 마취전문의성명 |
| 병록번호 | | | | | | | | | |

| 도착경로 | 입원경로 |
|---|---|
| 기타 | 외래 |

| 구분 | 코드 | 조정금액 |
|---|---|---|

| 구분 | 상병명 및 부위 | 분류기호 | 수술 | 과목 | 전문과목 | 상해 | 특정기호 | 요양개시일 | 진료결과 |
|---|---|---|---|---|---|---|---|---|---|
| | 둔부 힘줄염 | M760 | | 재활 | | | | 2007.06.01 | |
| | 상세불명의 윤활막염 및 건초염 | M659 | | 재활 | | | | 2007.06.01 | |
| | [redacted] | | | | | | | | |

| | |
|---|---|
| 감 I | |
| 감 II | |
| 중 I | |
| 중 II | |
| 계 | |
| 초진료 | |
| 재진료 | 11,880 |
| 처방횟수 | |

초진 야.공 | 재진 외부 4 야.공 | 요양일수 4 | 내원일자 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

| | | | |
|---|---|---|---|
| 1.초진 | | | |
| 재진 | | | 11,880 |
| 응급.회송등 | | | 160 |
| 2.일반 | | | |
| 내.정.소 | | | |
| 집중치료 | | | |
| 격리병실 | | | |
| 분만관련관리료 | | | |
| 기타 | | | |
| 3.내복 | | | |
| 외용 | | | |
| 처방전 | | | |
| 4.피하근육 | | | |
| 정맥내 | | | |
| 수액제 | | | |
| 특정재료 | | | |
| 기타 | | | |
| 5.마취료 | | | |
| 6.이학요법 | 1,380 | | 13,880 |
| 7.정신요법 | | | |
| 8.처치.수술 | | | |
| 수혈 | | | |
| 캐스트 | | | |
| 9.자체검사 | | | |
| 위탁검사 | | | |
| 10.진단 | | | |
| 치료 | | | |
| L.요양병원정액 | | | |
| S 특수장비 | | | |
| C CT | | | |
| M MRI | | | |
| 약품투약료 | | | |
| C T 총액 | | | |
| MRI 총액 | | | |
| 소계 | I | | 13,420 |
| | II | | 13,880 |
| 가산율 | 25% | 금 | 3,470 |
| 본인상한액 | | | |
| 급여비총액 | | | 30,770 |
| 본인부담금 | | | 15,380 |
| 장애 | | | |
| 대불 | | | |
| 청구액 | | | 15,390 |
| 100/100본인 | | | |
| 비급여총액 | | | |
| 진료비총액 | | | |

| 코드 | 분류명 및 품명 | 단가 | 1회 | 일투 | 총투 | 금액 | 비고 |
|---|---|---|---|---|---|---|---|
| | 01항. 진찰료 | | | | | | |
| AA222 | 재진50%-물리치료등 | 2,970.0 | 1.00 | 1.00 | 1 | 2,970 | 2007.06.01 |
| AA222 | 재진50%-물리치료등 | 2,970.0 | 1.00 | 1.00 | 1 | 2,970 | 2007.06.04 |
| AA222 | 재진50%-물리치료등 | 2,970.0 | 1.00 | 1.00 | 1 | 2,970 | 2007.06.05 |
| AA222 | 재진50%-물리치료등 | 2,970.0 | 1.00 | 1.00 | 1 | 2,970 | 2007.06.08 |
| AL601 | 외래환자 의약품관리료-1일분(종합병원) | 40.0 | 1.00 | 1.00 | 1 | 40 | 2007.06.01 |
| AL601 | 외래환자 의약품관리료-1일분(종합병원) | 40.0 | 1.00 | 1.00 | 1 | 40 | 2007.06.04 |
| AL601 | 외래환자 의약품관리료-1일분(종합병원) | 40.0 | 1.00 | 1.00 | 1 | 40 | 2007.06.05 |
| AL601 | 외래환자 의약품관리료-1일분(종합병원) | 40.0 | 1.00 | 1.00 | 1 | 40 | 2007.06.08 |
| | 06항. 이학요법료 | | | | | | |
| MM080 | 간섭파전류치료 | 2,600.0 | 1.00 | 1.00 | 1 | 2,600 | * 2007.06.01 |
| MM080 | 간섭파전류치료 | 2,600.0 | 1.00 | 1.00 | 1 | 2,600 | * 2007.06.04 |
| MM080 | 간섭파전류치료 | 2,600.0 | 1.00 | 1.00 | 1 | 2,600 | * 2007.06.05 |
| MM080 | 간섭파전류치료 | 2,600.0 | 1.00 | 1.00 | 1 | 2,600 | * 2007.06.08 |
| MM020 | 심층열치료 [1일당] | 870.0 | 1.00 | 1.00 | 1 | 870 | * 2007.06.01 |
| MM020 | 심층열치료 [1일당] | 870.0 | 1.00 | 1.00 | 1 | 870 | * 2007.06.04 |
| MM020 | 심층열치료 [1일당] | 870.0 | 1.00 | 1.00 | 1 | 870 | * 2007.06.05 |
| MM020 | 심층열치료 [1일당] | 870.0 | 1.00 | 1.00 | 1 | 870 | * 2007.06.08 |
| A00302601 | 264 로시덴겔/C 214003CCM | 69.0 | 1.00 | 5.00 | 1 | 345 | 2007.06.01 예외[55] |
| A00302601 | 264 로시덴겔/C 214003CCM | 69.0 | 1.00 | 5.00 | 1 | 345 | 2007.06.04 예외[55] |
| A00302601 | 264 로시덴겔/C 214003CCM | 69.0 | 1.00 | 5.00 | 1 | 345 | 2007.06.05 예외[55] |
| A00302601 | 264 로시덴겔/C 214003CCM | 69.0 | 1.00 | 5.00 | 1 | 345 | 2007.06.08 예외[55] |

◐ 특정내역 ◑

| 명일련 | 줄번호 | 특정코드 | 내역 |
|---|---|---|---|
| | 17 | JS002 | 55 |
| | 18 | JS002 | 55 |
| | 19 | JS002 | 55 |
| | 20 | JS002 | 55 |

---------------->> THE END <<----------------

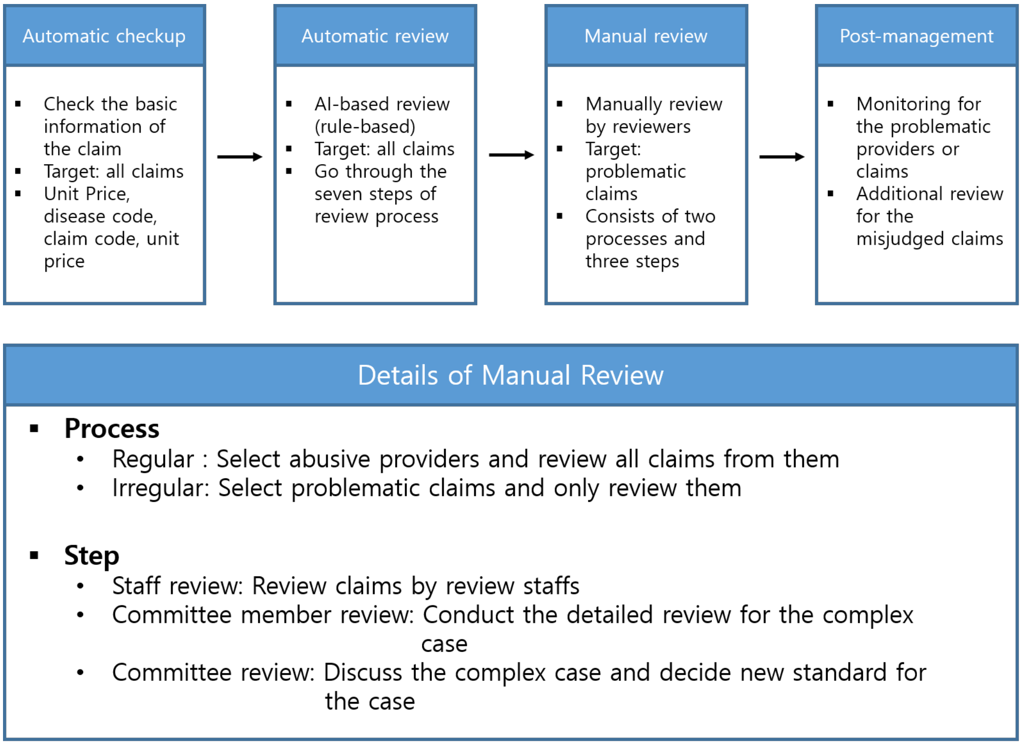
전문가점검 :
3001/동일수진자명세서(04172)(02287) 입(내)원일자 일부 중복 - 전문가점검 제외 명세서이므로 확인심사요

************** 심사결정사항 **************
심사차수 : 20070805 묶음번호 : 8390200 접수번호 : 4102235 명일련 : 0417200 수진자 : [redacted]
요양급여비용총액: 30770 본인환급금: 0
본인부담금: 15380 본인추가부담금: 0

Figure 2.5: An example of medical claim of South Korea

the precise examination by the committee members. We illustrate the review process of HIRA in Figure 2.6.

There are several databases that are separately stored within the HIRA data warehouse. Each database stores important information about the insurance claim such as claim information, treatment information, disease information, and review details. The details of the databases that we integrated into a single data are listed in Table 2.2. We did not use all data in the databases. Instead, we extracted records that are relative to the claim filed in 2016. Also, we selected several important variables in consultation with the field experts. Then, we integrated the tables into a single data and preprocessed the resulting table. As a result, we extracted about



Automatic checkup

- Check the basic information of the claim
- Target: all claims
- Unit Price, disease code, claim code, unit price

Automatic review

- AI-based review (rule-based)
- Target: all claims
- Go through the seven steps of review process

Manual review

- Manually review by reviewers
- Target: problematic claims
- Consists of two processes and three steps

Post-management

- Monitoring for the problematic providers or claims
- Additional review for the misjudged claims

Details of Manual Review

- **Process**
  - Regular : Select abusive providers and review all claims from them
  - Irregular: Select problematic claims and only review them
- **Step**
  - Staff review: Review claims by review staffs
  - Committee member review: Conduct the detailed review for the complex case
  - Committee review: Discuss the complex case and decide new standard for the case

Figure 2.6: The review process in HIRA

107 million treatment records.

Then, we selected 18 numerical variables and 18 categorical variables as input variables for modeling. We cannot list all variables that we used, because of the data confidentiality issue. However, we explain three important categorical variables that have high-cardinality: disease codes, special patient code, and the treatment code. In the raw data, the disease codes are in Korean standard classification of diseases (KCD) format. However, we used aggregated codes because there are too many codes to use all of them in modeling. The treatment code variable has also same characteristics. Because of the same reason, we used aggregated codes of treatment codes. Finally, each of them has 1902, 196, 7882 category values in modeling. In spite of this process, these variables still have high cardinality. So, we trained embedding vectors of these variables in modeling, as we presented in subsection 2.3.1.

The class to be predicted is defined as follows. When reviewers label certain treatment to be unnecessary for the patient, the abused amount of the treatment equals the claimed amount for the treatment. When a treatment is considered to be necessary but excessive, the abused amount is less than claimed amount. We define these treatments as abused treatments. Otherwise, we define a treatment with no abused amount as a normal treatment. The information about abused amount is stored in the 'Review details' database in Table 2.2. We report the number of providers, claims, treatments, and class ratio by department in Table 2.3. [112] Due to the data confidentiality issue, we anonymized the name of the department. For modeling and evaluation, we split the data set into train, validation, and test set by 6:2:2 with a similar class ratio. We use the train and the validation set for training the treatment scoring model for each department, and the test set for evaluation.

Table 2.2: Used databases and their details

| Database | Details |
|---|---|
| Claim information | - Basic information of claim<br>ex) claim number, patient information |
| Treatment details | - Treatment or prescription information<br>ex) treatment code, prescription code, daily dosage |
| Review details | - Manual review results<br>ex) review code, abused amount |
| Disease information | - Disease codes related to the claim<br>ex) main disease code, sub disease codes |

Table 2.3: Data statistics

| Department | Number of providers | Number of claims | Number of treatments | Proportion of abuse |
|---|---|---|---|---|
| A | 393 | 820,511 | 58,296,667 | 2.28% |
| B | 255 | 328,406 | 24,122,644 | 4.94% |
| C | 33 | 154,254 | 9,596,701 | 0.89% |
| D | 103 | 165,294 | 6,941,573 | 2.14% |
| E | 156 | 78,740 | 5,016,126 | 1.72% |
| F | 116 | 50,698 | 3,191,256 | 1.97% |

### 2.4.2 Experimental Settings

As we described in subsection 2.3.1, the treatment scoring model is a deep neural network with embedding layers for categorical variables with high cardinality. We create a non-linear decision boundary by activating hidden layers with non-linear activation functions, such as sigmoid, tanh, ReLU [64], ELU [17], LeakyReLU [57]. For categorical variables with high cardinalities, we used different embedding dimension in compliance with their cardinalities. Also, we experimented with various hidden layer size. To prevent overfitting, we used dropout [85] and early stopping techniques [74]. The maximum number of iterations is 200000 and the batch size is 1024. We also experimented with different optimizer such as Adagrad [24], RMSProp [89], and Adam [39]. In all cases, we set the initial learning rate at 0.0002.

Another important issue in this problem is the class imbalance problem. As we can see in Table 2.3, the class ratio is extremely imbalanced. The abuse cases occur rarely. If we do not address this issue properly, the neural network will learn the parameters so that the error is minimized only for the majority class data. Because the loss from the minority class is much less of importance than that of the majority class. In order to prevent this problem, we oversample the minority class data in every mini-batch.

We should also determine how to express categorical variables with high cardinality. The most common approach is to create dummy variables. However, as data grows, it may suffer from the curse of dimensionality [90]. In particular, these variables involves memory issues.

In this paper, we cope with such issues by using two methods. Firstly, we implement the proposed method with Tensorflow package [1]. We also used Compressed Sparse Row methods in Scipy package [91] to convert the dense matrix to sparse one. Then train a logistic regression model. In both cases, we select a model with the largest area under precision-recall curve (AUPRC) in the validation set [21]. We illustrate the process of selecting the best model in Figure 2.7.

### 2.4.3 Evaluation Measure (1): Relative Efficiency

In subsection 2.4.3 and 2.4.4, we elaborate on the devise on the evaluation measure. The baseline model for our experiments is the scoring model employed by HIRA. This model is based on discriminant analysis method with a set of provider-level variables. This model calculates the abuse scores for all providers. Then, reviewers select several suspicious providers based on the scores and examine all claims from them. Otherwise, the proposed method is based on a deep neural network with treatment-level variables. In this subsection, we explain a performance measure named relative efficiency, which quantifies the extent of efficiency improved by using the proposed method over the previous method.

The scatter plots on the left side of Figure 2.8 plot the abuse score against the total abused amount of each provider when evaluated by model A (above) and

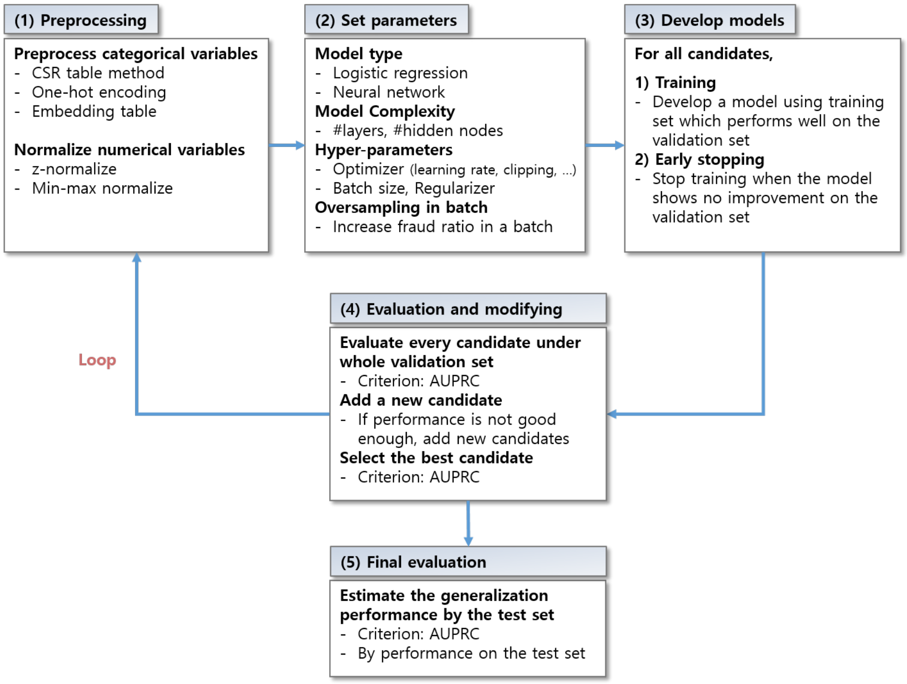

(1) Preprocessing
Preprocess categorical variables
- CSR table method
- One-hot encoding
- Embedding table
Normalize numerical variables
- z-normalize
- Min-max normalize
(2) Set parameters
Model type
- Logistic regression
- Neural network
Model Complexity
- #layers, #hidden nodes
Hyper-parameters
- Optimizer (learning rate, clipping, ...)
- Batch size, Regularizer
Oversampling in batch
- Increase fraud ratio in a batch
(3) Develop models
For all candidates,
1) Training
- Develop a model using training set which performs well on the validation set
2) Early stopping
- Stop training when the model shows no improvement on the validation set
Loop
(4) Evaluation and modifying
Evaluate every candidate under whole validation set
- Criterion: AUPRC
Add a new candidate
- If performance is not good enough, add new candidates
Select the best candidate
- Criterion: AUPRC
(5) Final evaluation
Estimate the generalization performance by the test set
- Criterion: AUPRC
- By performance on the test set

Figure 2.7: Training and selecting the best treatment scoring model

model B (below). If the scoring model is well-trained, the model will assign a high score to an abusive provider. That is, in other words, the model score and the actual abused amount should increase in proportion. The scatter plots on the left side of Figure 2.8 shows that the model B is better trained than model $A$. In the meantime, the right panel of Figure 2.8 reports the actual cumulative abused amount in descending order of each model's scores. Suppose that only half of all providers have been examined for abuse cases. According to the right-hand side plot, Model $B$ has detected approximately 80% of the entire pool of abused amount, while Model $A$ has only detected about 50% of abused amount. From this graphical investigation, we can infer that model $B$ examines claims more efficiently than model $A$ does.

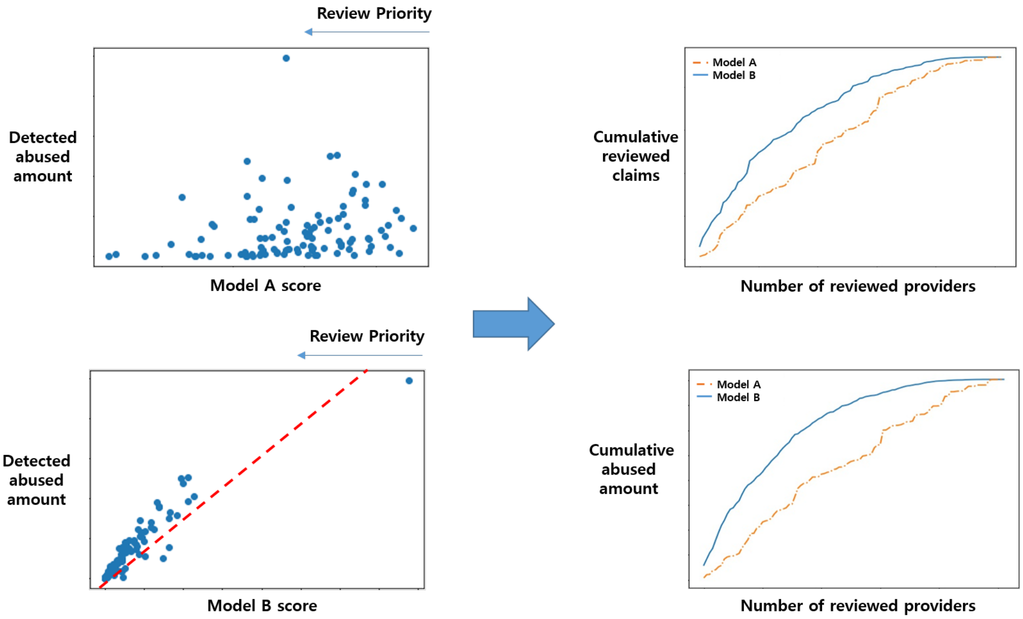


Figure 2.8: The concept of relative efficiency

From now on, we establish the definition of the efficiency of the reviewing process more concretely. First, we define the efficiency of review as the abused amount detected in relation to the efforts required for the examination. Until now, we have regarded the number of examined providers as the efforts. However, this is not enough. Suppose there are two providers, where one submits more claims than the other. In this case, greater efforts are required to review all the claims filed by the former than those by the latter. In other words, the amount of effort to review all the claims varies from provider to provider depending on the number of filed claims. This is the reason why we have to define the efforts as the number of examined claims rather than the number of examined providers. Therefore, in order to quantify the efficiency, we consider both the cumulative number of examined claims and the cumulative abused amount as illustrated in the Figure 2.8.

Mathematically, we express the efficiency of review as follows. Suppose there are $N$ providers for a department, and the number of claims and the number of medical treatments are defined as $m_1, m_2, \ldots, m_N$ and $n_1, n_2, \ldots, n_N$, respectively. Further, suppose that all providers are sorted in the descending order by the abuse score calculated from model A. The number of claims and treatments can then be represented by $m^A_{(1)}, m^A_{(2)}, \ldots, m^A_{(N)}$ and $n^A_{(1)}, n^A_{(2)}, \ldots, n^A_{(N)}$, respectively. In addition, we define $d^A_{(1)}, d^A_{(2)}, \ldots, d^A_{(N)}$ as the abused amount detected for each provider. Then, for providers with the top-$k$ highest scores, the number of claims, treatments and the abused amount are represented by $m_k$,$n_k$ and $d_k$. More specifically, the total number of claims is $M = m_1 + m_2 + \ldots + m_N = m^A_{(1)} + m^A_{(2)} + \ldots + m^A_{(N)}$. Now suppose the reviewers can screen $p\%$ of the total number of claims. That is, in other words, the reviewers can only screen $0.01pM$ claims. Then there exists $h$ that satisfies the following inequalities.

$$\sum_{i=1}^{h} m^A_{(i)} \le 0.01pM, \quad \sum_{i=1}^{h+1} m^A_{(i)} \ge 0.01pM$$

Here, the detected abused amount is $\sum_{i=1}^{h} d^A_{(i)}$. The efficiency of review is now defined as the total abused amount detected by the reviewer in comparison to the number of reviewed claims. If the reviewers select providers with scores computed by Model A, our proposed model, the efficiency can be represented as follows:

$$e^A_p = \frac{\sum_{i=1}^{h} d^A_{(i)}}{\sum_{i=1}^{h} m^A_{(i)}}$$

We are not at liberty to compute this measure explicitly due to data confidentiality issues. Hence, we replace it with the following term, called relative efficiency, to compare the efficiency of the two scoring models. Mathematically, relative efficiency can be expressed as below:

$$e_p^{A,B} = \frac{e_p^A}{e_p^B}$$

This measure quantifies improvements in efficiency improvement when selecting providers to review with model A as the base for comparison.

The number of providers reviewed may change at every review session. Hence, it is essential to be able to compute efficiency even though the size of providers are varying. We incorporate this idea into the relative efficiency measure and redefine the term as follows.

$$e_p = \frac{e_p^{proposed}}{e_p^{HIRA}}$$

Here, $HIRA$ stands for the previous scoring model that HIRA has been using, and *proposed* stands for the proposed scoring model.

### 2.4.4 Evaluation Measure (2): Precision at $k$

The concept of precision at $k$ refers to the proportion of relevant items in the top-$k$ item set retrieved. It is widely used in information retrieval field to measure the performance. In this study, we re-define the precision at $k$ measure to fit our purpose as follows. Suppose $A_k$ as the set of the institutions with top-$k$% abuse score, and $B_k$ as the set of the providers with top-$k$% abused amount. Let the precision at $k$ represent the proportion of the providers with top-$k$% abused amount in the providers to top-$k\%$ abuse score institution set. Then, mathematically, precision at $k$ can be expressed as below:

$$Pr_k = \frac{|A_k \cap B_k|}{|A_k|}$$

This metric measures the model's ability to detect providers with greater abused amount. In other words, it measures the extent of the model's ability to detect providers with a severely abusive billing pattern.

## 2.5 Results

### 2.5.1 Results in the test set

We illustrate the change of cumulative abused amount at different portions of the reviewed claims at department A in Figure 2.9. Suppose the reviewers can only examine 80% of the total claims. If they select the providers for review based on the score of the previous model, they will select 340 providers to review. In contrast, they will select 220 providers with the proposed model. If they reviewed all claims from 220 providers that the proposed model has recommended, they can detect 1.09 times more abused amount than reviewing all claims from the 340 providers recommended by the previous model. In short, they proposed model is 1.09 times more efficient than the previous model at the 80% level. Similarly, the proposed model is 1.13 times more efficient than the previous model at the 60% level, and 1.26 times more efficient than the previous model at the 40% level. We report relative efficiency values at various levels of proportions of claims reviewed for each department in Table 2.4. In Table 2.4, 'max' means the level when the maximum relative efficiency is achieved. We can see that the relative efficiency is larger or equal to 1 in most cases. It means the proposed model is more efficient than the previous model in most cases.

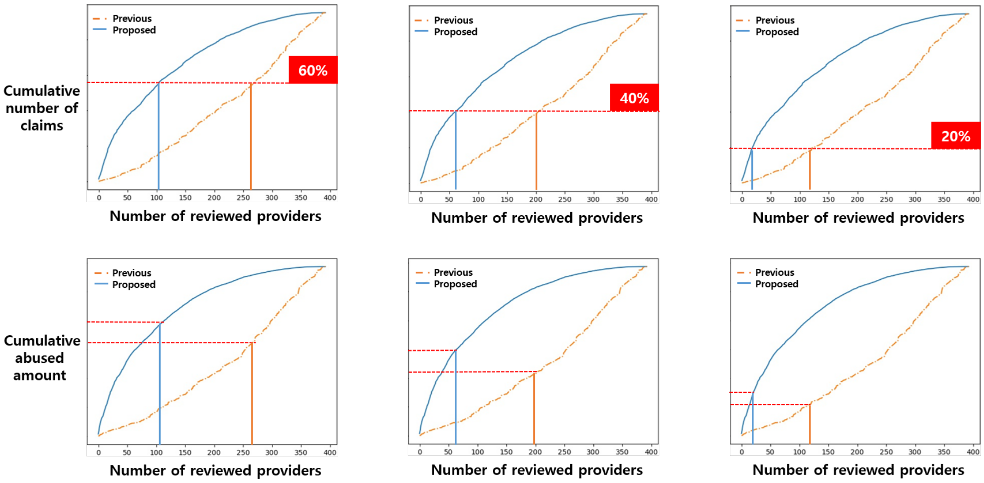


Figure 2.9: Relative efficiency at different levels

One more thing that we can see from Table 2.4 is the tendency that the relative efficiency values tend to grow larger at small $p$ than the larger one. This implies that the proposed model assigns higher scores to the more highly abusive providers, while previous model fails to do so. It is clearly observed in Table 2.5, that at small $k$, the precision at $k$ of the proposed model shows much better performance than that of the previous model.

Table 2.4: Relative efficiency on the test set by department

| Department | $e_{20\%}$ | $e_{40\%}$ | $e_{60\%}$ | $e_{80\%}$ | $e_{MAX}$ |
|---|---|---|---|---|---|
| A | 1.03 | 1.28 | 1.13 | 1.09 | 1.33 |
| B | 3.33 | 1.91 | 1.26 | 1.14 | 3.50 |
| C | 1.95 | 2.10 | 2.10 | 1.19 | 2.10 |
| D | 1.24 | 1.13 | 1.10 | 1.19 | 1.50 |
| E | 1.61 | 1.23 | 1.21 | 1.17 | 1.61 |
| F | 0.87 | 1.18 | 1.09 | 1.23 | 1.76 |

Table 2.5: Precision at $k$ on the test set by department

| Department | $Pr_{10}$ | | $Pr_{20}$ | | $Pr_{30}$ | | $Pr_{40}$ | | $Pr_{50}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* |
| A | 0.00 | **0.70** | 0.05 | **0.82** | 0.16 | **0.84** | 0.24 | **0.92** | 0.37 | **0.90** |
| B | 0.00 | **0.77** | 0.06 | **0.90** | 0.08 | **0.88** | 0.14 | **0.87** | 0.29 | **0.92** |
| C | 0.00 | **0.75** | 0.43 | **1.00** | 0.30 | **1.00** | 0.50 | **0.93** | 0.65 | **0.94** |
| D | 0.09 | **0.73** | 0.43 | **0.71** | 0.42 | **0.90** | 0.57 | **0.91** | 0.64 | **0.90** |
| E | 0.38 | **0.94** | 0.38 | **0.81** | 0.47 | **0.87** | 0.51 | **0.95** | 0.59 | **0.91** |
| F | 0.25 | **0.75** | 0.46 | **0.79** | 0.51 | **0.91** | 0.55 | **0.87** | 0.60 | **0.88** |

### 2.5.2 The Relationship among the Claimed Amount, the Abused Amount and the Abuse Score

The proposed model calculates the abuse score of the provider by summing up the resulting scores from multiplying the output of the treatment scoring model and claimed amount for each treatment. By definition, the abuse score of a provider is variant to the total claimed amount filed by the provider. Without the scores resulting from the scoring model, the abuse score would merely reflect the total claimed amount from the provider. It means that the model only selects the providers with top-$k$ highest claimed amount. However, by including results from the treatment scoring model, the proposed model selects broader types of abuse cases.

This is illustrated in Figure 2.10. It shows the total abused amount and the abuse scores 20 providers with the largest claimed amount. The providers are sorted in descending order by the claimed amount. First, let us look at the relationship between the claimed amount and the abused amount. Providers are sorted in descending order by claimed amount, but the abused amount does not tend to descend. It means that large claimed amount does not mean large abused amount. Likewise, the abuse score does not tend to descend, which means that the score is not simply proportional to the claimed amount. The difference is from the result which is calculated from the treatment scoring model. Due to this term, the bias caused by the claimed amount is reduced. Also, we can see that abuse score moves in accordance with the real abused amount. This means that the abuse score calculated by the proposed model estimate the abuse degree of the provider well.

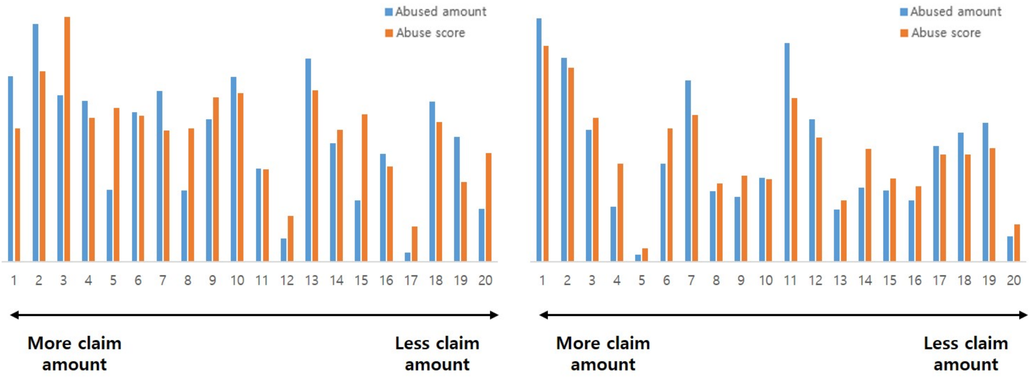


Figure 2.10: The relationship among claimed amount, abused amount and proposed abuse score

### 2.5.3 The Relationship between the Performance of the Treatment Scoring Model and Review Efficiency

In this subsection, we will discuss the performance of the treatment scoring model on the performance of the provider scoring model. In the previous subsection, we built lots of treatment scoring models with various hyper-parameters and select a model that has the best performance in the validation set.

In order to show the impact of the treatment scoring model on the performance of the provider scoring model, we performed the following experiment. First, randomly select a treatment scoring model whose performance is slightly lower than the best one. Then, calculate the performance of the provider scoring model with both the selected one and the best one. We show the impact indirectly by comparing them. We report the performances of both cases in Table 2.6 and Table 2.7. In most cases, the performance of the selected model is slightly less than or similar to the best one. In particular, the relative efficiency is seemed to be very similar. However, remind the relative efficiency is a 'relative' performance measure. There are cases that the difference in relative efficiency by 0.1 means millions of dollars. Therefore, it is not a small difference.

Table 2.6: Relative efficiency of the randomly chosen model and the best model

| Department | $e_{20\%}$ | | $e_{40\%}$ | | $e_{60\%}$ | | $e_{80\%}$ | | $e_{MAX}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* |
| A | **1.05** | 1.03 | 1.22 | **1.28** | 1.10 | **1.13** | 1.06 | **1.09** | 1.25 | **1.33** |
| B | **3.07** | 1.33 | 1.84 | **1.91** | 1.25 | **1.26** | 1.13 | **1.14** | 3.43 | **3.50** |
| C | 1.95 | **1.95** | 2.10 | **2.10** | 2.10 | **2.10** | 1.19 | **1.19** | 2.10 | **2.10** |
| D | 0.98 | **1.24** | 0.89 | **1.13** | 1.03 | **1.10** | 1.04 | **1.19** | 1.33 | **1.50** |
| E | 1.52 | **1.61** | 1.22 | **1.23** | 1.20 | **1.21** | 1.16 | **1.17** | **1.56** | 1.61 |
| F | 0.64 | **0.87** | 0.98 | **1.18** | 1.09 | **1.09** | 1.22 | **1.23** | 1.22 | **1.76** |

### 2.5.4 Treatment Scoring Model Results

In this subsection, we will discuss how the structure of the treatment model affects the performance of the proposed model. In the previous subsections, we compared the proposed method to previous method that is based on the provider-level variables.

Table 2.7: Precision at $k$ of the randomly chosen model and the best model

| Department | $Pr_{10}$ | | $Pr_{20}$ | | $Pr_{30}$ | | $Pr_{40}$ | | $Pr_{50}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* | *Rand* | *Best* |
| A | 0.68 | **0.70** | 0.79 | **0.82** | 0.81 | **0.84** | 0.87 | **0.92** | 0.87 | **0.90** |
| B | **0.81** | 0.77 | 0.84 | **0.90** | 0.86 | **0.88** | 0.84 | **0.87** | 0.90 | **0.92** |
| C | 0.75 | **0.75** | 0.86 | **1.00** | 0.90 | **1.00** | 0.93 | **0.93** | 0.94 | **0.94** |
| D | 0.64 | **0.73** | 0.62 | **0.71** | 0.81 | **0.90** | 0.81 | **0.91** | 0.87 | **0.90** |
| E | 0.88 | **0.94** | 0.75 | **0.81** | **0.89** | 0.87 | 0.92 | **0.95** | 0.91 | **0.91** |
| F | 0.75 | **0.75** | **0.83** | 0.79 | 0.89 | **0.91** | 0.83 | **0.87** | 0.86 | **0.88** |

However, since we exploit the treatment-level variables in the proposed method, it is not appropriate to compare directly between two models. Instead, we compared the proposed model to logistic model that uses treatment-level variables. By this comparison, the proposed structure is appropriate for the treatment scoring model.

In order to handle the categorical variables with high cardinality, we applied CSR method provided by the Scipy package. Also, we experimented with various class weight in training logistic models. Then, we selected a model that shows the best performance in the validation set.

In Table 2.8, we reported the AUPRCs from the logistic regression and the proposed treatment scoring model for each subject in the test set. As we can see, the proposed model performs much better than the logistic regression model in every case. For the case of logistic regression, categorical variables with high cardinality are one-hot encoded. So, the dimension of the data becomes much larger than before. As a result, it requires much more complex computation while the performance does not meet with the complexity of the model. However, the proposed model learns not only the network parameters but also the embedding function to minimize the error. This might be a reason that led to much better performance as compared to the logistic regression model.

In subsection 2.5.3, we have shown that the performance of the treatment scoring model affects the performance of the provider scoring model. From this result, both model complexity and learning the embedding function of categorical variables with high cardinality play an important role in determining the performance of the model.

Table 2.8: The AUPRC of the best treatment scoring model

| Department | Logistic regression | Proposed model |
|---|---|---|
| A | 0.24 | **0.60** |
| B | 0.41 | **0.72** |
| C | 0.44 | **0.73** |
| D | 0.31 | **0.63** |
| E | 0.30 | **0.69** |
| F | 0.25 | **0.63** |

### 2.5.5 Post-deployment Performance

Suppose a situation that reviewers select abusive providers from claim data in previous year and examine all claims from them in this year. If the scoring model performs well and the data distribution is similar between two years, the reviewing process may be efficient. If then, the proposed model can be used in reality. So, we experimented with the claim data filed in 2016 and 2017. We trained models and selected abusive providers with the claim data in 2016. Then, we evaluated the performance

Table 2.9: Data statistics used for evaluating post-deployment performance

| Department | Number of institutions | Number of claims | Number of treatments | Proportion of abuse |
|---|---|---|---|---|
| A | 259 | 280,083 | 24,274,388 | 0.61% |
| B | 101 | 77,247 | 5,875,745 | 0.64% |
| C | 24 | 14,706 | 1,029,135 | 0.63% |
| D | 76 | 75,922 | 3,646,965 | 0.52% |
| E | 128 | 31,214 | 2,121,534 | 0.49% |
| F | 90 | 28,902 | 1,915,349 | 0.67% |

with the data in 2017. There may some providers that exists in 2017 but not in 2016 and vice versa. We excluded such providers in the experiment. Table 2.9 lists the summary statistics that we used. We report the performance in Table 2.10 and Table 2.11. As we can see from these tables, the proposed model is more efficient than the previous model. Also, it detects abusive providers well.

Table 2.10: Relative efficiency in 2017 based on the results of 2016

| Department | $e_{20\%}$ | $e_{40\%}$ | $e_{60\%}$ | $e_{80\%}$ | $e_{MAX}$ |
|---|---|---|---|---|---|
| A | 1.37 | 1.32 | 1.12 | 1.09 | 1.38 |
| B | 5.95 | 2.10 | 1.43 | 1.11 | 6.10 |
| C | 1.00 | 3.45 | 2.10 | 1.70 | 3.53 |
| D | 1.60 | 1.14 | 1.21 | 1.13 | 1.80 |
| E | 1.41 | 1.26 | 1.22 | 1.22 | 1.43 |
| F | 0.82 | 0.99 | 0.98 | 1.15 | 1.39 |

Table 2.11: Precision at $k$ in 2017 based on the results of 2016

| Department | $Pr_{10}$ | | $Pr_{20}$ | | $Pr_{30}$ | | $Pr_{40}$ | | $Pr_{50}$ | |
|---|---|---|---|---|---|---|---|---|---|---|
| | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* | *Pre* | *Pro* |
| A | 0.00 | **0.65** | 0.06 | **0.71** | 0.14 | **0.72** | 0.24 | **0.78** | 0.37 | **0.81** |
| B | 0.00 | **1.00** | 0.00 | **0.76** | 0.00 | **0.81** | 0.07 | **0.88** | 0.24 | **0.82** |
| C | 0.33 | **0.33** | 0.40 | **0.60** | 0.38 | **0.88** | 0.50 | **0.80** | 0.58 | **0.92** |
| D | 0.00 | **0.50** | 0.31 | **0.69** | 0.39 | **0.65** | 0.45 | **0.84** | 0.63 | **0.84** |
| E | 0.39 | **0.69** | 0.35 | **0.77** | 0.46 | **0.74** | 0.48 | **0.71** | 0.58 | **0.80** |
| F | 0.11 | **0.56** | 0.39 | **0.78** | 0.57 | **0.82** | 0.59 | **0.81** | 0.58 | **0.78** |

## 2.6 Summary

Healthcare insurance companies manually review all the medical claims to detect abuse in order to avoid issuing unnecessary compensations. However, as the number of claim filings grow exponentially, the cost of manual review increases astronomically, which calls for a more efficient review process. By efficiency, we set our objectives to detect as much abused amount correctly as possible with minimum effort. It is particularly important to effectively screen out abusive medical providers, as they are more likely to prescribe unnecessary treatments to the patients. Such a screening process, in turn, will require a scoring scheme which that measures the degree of abuse.

In this chapter, we propose the very first model that scores abusive billing patterns of providers using the medical treatment data. The proposed model consists of two steps: (1) training a neural network to compute the likelihood of abuse for each treatment, and (2): calculating the abuse score for each treatment and aggregating the results up to the provider level. The abuse score for each treatment is calculated by multiplying the neural network result with claimed amount. Experiment results show that our proposed model scores abusiveness better than the model with features summarized at the provider-level.

The main contribution of this chapter lies in that it is one of the first research detecting the abusive provider using medical treatment data, which is the finest-grained level data in terms of medical claims. Previous studies extract the provider-level variables such as the number of prescriptions per day or the average cost per claim and use these variables for training. This way, the model cannot properly account for information apparent only at the claim or treatment-level. In contrast, we fully exploit the fine granularity of the treatment data to train the model. The experiment results show that the proposed model performs better than the model with provider-level variables. In addition, we devise performance metrics, relative efficiency and precision at k, to quantify the efficiency improvement. Using these metrics, we show that the reviewers can review more efficiently by looking at providers determined to be suspicious of abuse by the proposed model as compared to examining those selected by the model with provider-level variables. Finally, we show that the performance of the treatment scoring scheme is important to computing an effective abuse score. This implies that training better neural network results in better performance.

If a provider is chosen to be abusive and all its claims are reviewed, it will not be reimbursed for the amount determined to be abused, which will result in the loss of the provider. Consequently, the provider does not want to be selected, which in turn reduces the waste of health insurance, so that insurance companies can reduce unnecessary costs. However, as the medical environment continues to change, it also creates forms of abuse that did not exist before. Previous scoring methods using the existing provider-level variables cannot adapt to this changing pattern of abuse. On the contrary, the proposed model scores abusivesness while adapting to changes in abuse patterns through regular retraining.

There are two potential limitations to our model. Firstly, we assume that the filed claims are uniformly distributed across time. Our experiment splits the entire data set into the training, validation, and test sets, of which the underlying assumption is that learning is not contingent upon time. However, from the practical point of view, such an assumption may not hold true for some cases. In the next chapter, we address this issue in greater detail and propose a model which accounts for seasonality.

Another limitation of the current model is that it does not explicitly consider the association relationship between diseases and treatments, one of the most significant factors in reviewing claims. In chapter 4, we discuss this issue in detail and propose a model explicitly dealing with this relationship.

# Chapter 3

# Detection of overtreatment by Diagnosis-related Group with Neural Network

## 3.1 Background

In chapter 2, we introduce the very first method, to our best knowledge, to detect abusive providers by using medical treatment data, which is the lowest level of healthcare information available. We show the review process would be more efficient if the field reviewers give priority to the candidates of abusive providers selected by the proposed model instead of those screened by the previous method. The proposed model computes the degree of the provider's abusiveness numerically, which helps interpret the detection result. The key assumption underlying our model is that the distribution of claim data is similar between the training set and the test set.

Before we discuss this issue, let us define the distribution of the claim data. We believe that the most important information from the claim filings is diseases of diagnose and the prescribed treatments. In the perspective of our data, then we consider a claim as a single value from a distribution of claims. Under this setting, we assert that the representative value of a claim should include both disease and treatment information. Here, every disease and treatment information does not have to be included. We can represent the claim by main disease and several important treatments only. Given these three requirements, we believe that the diagnosis-related group (DRG) is an appropriate measure to serve as the aforementioned representative value. It includes information about the main disease the patient is diagnosed of, as well as the treatments the practitioner has prescribed.

The previous model used by HIRA assumes that claims data follows a homogeneous distribution, which is too strict of an assumption to assert to hold true in reality. One well-known counterexample is seasonality. A handful of diseases, flu, for example, show period surge of infected patients for a specific period of time, a characteristic to which we refer as seasonality. Seasonality implies that the distributions of claims may shift by time, according to its seasonal surge and dissolution. In fact, as we can see by observing Figure 3.1 that this assumption is realistic.

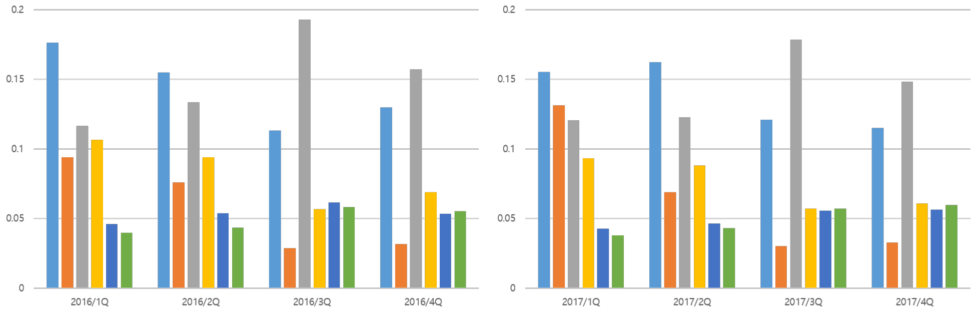


Figure 3.1: The distribution of the patient group in a department

Suppose we train the model ignoring these seasonal patterns. The model's primary objective to minimize the training error; consequently, the model will focus solely on claim data whose DRG-code is the majority. Hence, the model won't perform well if it is trained to learn to compute the degree of abuse under the homogeneity assumption. A candidate solution is to train the model using observations from the same point of time every year. For example, one may choose to use claims filed in during the 1st financial quarter of last year for training and then use claims the 1st quarter of this year to estimate the score for the degree of abuse. Such an approach is still not safe from events that randomly or unexpectedly taking place, such as medical inventions and innovations. Suppose an ingenious clinical innovation has intervened between the training period and the scoring period. Then, information used for training is outdated and there are new forces governing the features of claims filed in this year, leading the model to poor performance.

In order to account for seasonality during learning, we train our treatment classification model with claims data grouped by DRG code. The DRG system is a type of patient classification scheme (PCS) which provides a means of relating the type of patients a hospital treats to the costs incurred by the hospital [11]. DRG system categorizes patient episodes by controlling the fundamental variations, which are assumed to be always present, among patients. Claims with the same DRG code include similar disease or treatment.

If we train models by DRG code, then the distribution of claims will be more homogeneous as compared to the existing model. Even if distinct seasonal characteristics, peculiar to each disease, exist, because the model is trained by similar diseases. Consequently, our model will produce results robust to seasonality. Suppose we have to detect abuse in medical treatments in department $A$. Also, suppose every claim in the department has one of two DRG codes: $D_a, D_b$. In the training set, the number of claim data with the DRG code $D_a$ is much larger than that of $D_b$. In this case, the model will be trained to minimize the training error from the data with the DRG code $D_a$. It means the training error from the data with the

DRG code $D_b$ is ignored relative to that with the DRG code $D_a$. However, suppose the number of claim data with DRG code $D_b$ is much larger than that of $D_a$ in the test data. In this case, the trained model cannot classify the data with DRG code $D_b$ and the performance of the model will decrease. However, if models are trained separately for DRG code $D_a$ and $D_b$, the performance will not degrade since data distribution in each data set is similar between the training set and the test set.

In this chapter, we propose to run the treatment classification model by DRG code unit. If then, the model can classify the treatment robust to seasonality. The DRG system has been used in patient classification. It has also been serving as the unit of the DRG-based payment system and as the standard of comparing medical institutions. Our work show the possibility that the DRG system can be used in the review process in healthcare insurance.

The rest of the chapter is organized as follows. In section 3.2, we introduce seasonality in disease and the concept of the DRG system. Section 3.3 provides detailed descriptions of the proposed model. In addition, we introduce strategies to compare performance between our model and the method that is suggested by Lee et al. [47]. In section 3.4, we elaborate on experiment settings. We also provide detailed description of the data and the preprocessing steps in this section. Section 3.5 reports experiment results. Finally, section 3.6 concludes the paper.

## 3.2 Literature review

### 3.2.1 Seasonality in disease

In public health, seasonality is a feature characterized by the surge of a certain disease recurring at a particular time period ([28], [58]). A variety of infectious diseases, such as influenza, as well as some respiratory diseases which are non-infectious, exhibit seasonality.

Even though the awareness for seasonality has existed for a while in the research field, the underlying mechanism of seasonality has not been fully explained. Clear understanding of seasonality will certainly prove beneficial for public health in many different aspects. Fisman [28] claimed that there are four major benefits that may rise from understanding the full mechanism of seasonality: *(1) improved understanding of host and pathogen biology and ecology, (2) enhanced accuracy of surveillance systems, (3) improved ability to predict epidemics and pandemics, (4)better understanding of the long-term implications of global climate change for infectious disease control.* To shed more realistic light on the potential benefits, we take the example of the two viral respiratory illnesses: severe acute respiratory syndrome (SARS) and coronavirus disease 19 (COVID-19). These two diseases are quite similar in a sense that their main agent of contagion is the coronavirus, which is a type of an enveloped RNA virus. If seasonal features associated with the spread of SARS were fully characterized, the results of which may serve as the basis to infer/predict the seasonality of COVID-19. Then, resources may have been allocated accordingly to detect and prevent the disease in a timely manner.

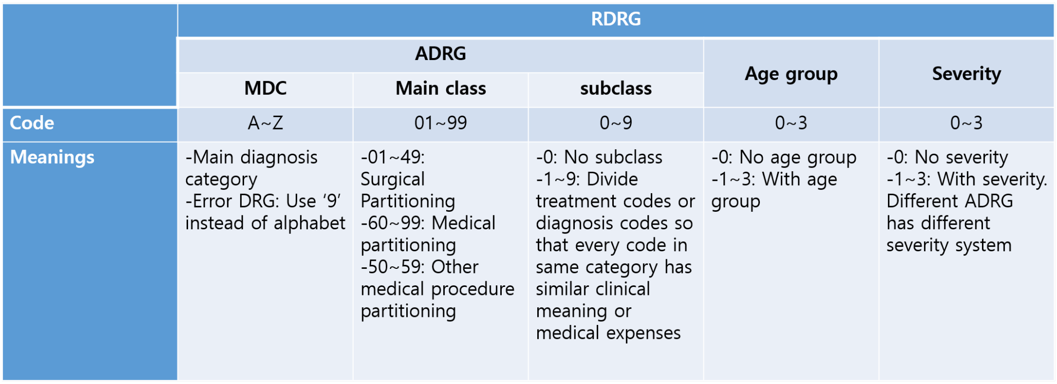
### 3.2.2 Diagnosis related group

Diagnosis-related group (DRG) is one brank of the patient classification system (PCS), which classifies patients in perspective of clinical records and medical resource consumption patterns such as diagnosis, procedures, or functional status [11]. It was first devised in Yale University in the late 1960s. Originally, the objective of

DRG was to create an efficient method for monitoring the quality of patient care and the utilization of service for each hospital. Additional adjustments were continuously made to the system since its first invention, raising its quality to the current level. Now, DRG is exploited in various ways, including hospital-to-hospital comparisons, patient classification, and evaluation of medical institutions. At the same time, it is also used as a unit of the bundled-payment system for healthcare insurance. Bundled-payment system is known to compensate for the shortcomings of the fee-for-service payment system which has been popular of choice. Under the fee-for-service payment system, the insurance company must reimburse for all the treatments provided to the patient. It is more likely to lead to over-treatment, since the provider can enjoy greater reimbursement by performing additional procedures. In contrast, under the bundled-payment system, each patient is classified by the DRG code, and the insurance company only has to compensate for the amount predefined for the subject patient category. In other words, regardless of the number of treatments performed by the provider, the insurance company compensates only for the pre-determined, fixed amount. This, by design, deters providers from over-treatment.

The Korean diagnosis-related group (KDRG) is a modified version of DRG, adjusted to reflect the peculiarity in the medical practice in Korea [38]. The first version of KDRG, KDRG v1.0, was first devised in 1986. Now it is updated to KDRG v4.3 with 2,753 codes for classifying the patients. These codes are constructed by combining Korean classification of diseases (KCD) and treatment codes.

The formation of the DRG code begins by splitting up all the principal diagnoses available into 23 main diagnostic categories (MDC). Then, the MDCs are subdivided either into medical or to surgical categories. For example, a patient is classified as surgical if the prescription on his/her claim includes surgery/operations. Otherwise, the patient is classified as a medical case. Surgical cases are further divided into the groups of finer granularity based on the precise surgical approaches performed; medical cases, based on the exact principal diagnosis. The DRG code assigned by this process is called as the Adjacent DRG (ADRG). In order to classify patients as accurately and appropriately as possible, the age group, as well as the complication and comorbidity factors, are also considered as the classification criteria. The final DRG code resulting after the whole process is called the Refined DRG (RDRG) code. The summary of the KDRG structure is shown in Figure 3.2.



| | RDRG | | | | |
|---|---|---|---|---|---|
| | ADRG | | | Age group | Severity |
| | MDC | Main class | subclass | | |
| Code | A~Z | 01~99 | 0~9 | 0~3 | 0~3 |
| Meanings | -Main diagnosis category<br>-Error DRG: Use '9' instead of alphabet | -01~49: Surgical Partitioning<br>-60~99: Medical partitioning<br>-50~59: Other medical procedure partitioning | -0: No subclass<br>-1~9: Divide treatment codes or diagnosis codes so that every code in same category has similar clinical meaning or medical expenses | -0: No age group<br>-1~3: With age group | -0: No severity<br>-1~3: With severity. Different ADRG has different severity system |

Figure 3.2: The structure of KDRG

## 3.3 Proposed method

This section details the structure of the proposed model which leans to classify the entered treatment to be normal or not. In this study, we use the treatment data, grouped by DRG code, for training as well as for inference. This approach distinguishes our proposed model from the treatment scoring model, as found in Lee et al. [47], which grouped treatment data by the practice department.

### 3.3.1 Training a deep neural network model for treatment classification

The proposed model employs a neural network structure through which a given treatment classified to be normal or to be abused. The input data for the model is heterogeneous, containing both numerical and categorical information. Numerical variables include the unit price of the treatment or the amount of dosage per day, while gender, age group, or the associated treatment codes are the examples of the categorical variables. In order to make the best use of such data as a valid input to a neural network, categorical information must be represented in a form of a numerical vector. One of the most common approaches is to one-hot encode by the given categories. It is not, however, an appropriate approach in this case, because there exist some categorical variables that are of a high-cardinality. If these variables are one-hot encoded, the dimension of the data would explode, hence the suffer from the curse of dimensionality. In our model, we rely on an embedding function, instead, to represent these heterogeneous variables as a vector. Our proposed model trains the embedding function during the training phase. Classification error is back-propagated to the embedding layers as well as the hidden layers.

We describe our model mathematically as follows. Define a medical treatment $t^{'} = [n_1, n_2, \ldots, n_k, c_1, c_2, \ldots, c_l]$, where $n_i$ represents the value of the numerical variable $v_i$ and $c_j$ represents the value of categorical variable $v_j$. We define $\boldsymbol{d}_j$ as the one-hot encoding vector of $c_j$. We represent a categorical variable $v_j$ with value $c_j$ as $\boldsymbol{x}_j = \boldsymbol{d}_j^T$. Otherwise, we compute an embedding vector for the corresponding categorical variable of high-cardinality, represented by $\boldsymbol{x}_j = \boldsymbol{d}_j^T \boldsymbol{W}_j$. Here, $\boldsymbol{W}_j$ stands for the embedding function of the categorical variable $v_j$.

Another important candidate of the heterogeneous input variables, which calls for extra-care, is the multi-valued categorical variable. A multi-valued categorical variable is defined as a variable that has more than one values for each entity. In our study, the major case of the multi-valued categorical variable may be found where a patient is diagnosed to carry more than one diseases. Such cases may be discovered by looking up cases where a practitioner prescribes treatments that may be associated with all the diseases the patient may be suffering from. Since there does not exist the grounds for the identification of the casual relationship between the rich variety of symptoms and the diseases causing these symptoms, as well as the effect of the prescription of the treatments to the corresponding symptoms, it is highly likely to prescribe and practice treatments in response to as many as the candidates of the diseases the subject patient may carry. Such a practice give rise to the multi-valued categorical variables in the input data. In order to effectively represent these variables as numerical vectors, we first embed them through our embedding model and then average the resulting embedding vectors by disease category

We express aforementioned process mathematically as follows. Suppose a given categorical variable $v_j$ is a multi-valued categorical variable. That is, a medical treatment variable $t^{'}$ is represented by $[n_1, n_2, \ldots, n_k, c_1, c_2, \ldots, [c_{j1}, c_{j2}, \ldots, c_{jm_t}], \ldots, c_l]$, where $[c_{j1}, c_{j2}, \ldots, c_{jm_t}]$ is the value of the multi-valued categorical variable $v_j$ for the corresponding treatment $t$. Here, $m_t$ represents the number of values in the variable. It is different by each treatment. Then, we compute the embedding vector for $k$-th value in multi-valued categorical variable and denote it as $\boldsymbol{x}_{jk} = \boldsymbol{d}_{jk}^T \boldsymbol{W}_j$.

Finally, the embedding vector is averaged by the disease category, which is denoted as follows.

$$\boldsymbol{x}_j = \frac{1}{m_t}\sum_k \boldsymbol{x}_{jk} = \frac{1}{m_t}\sum_k \boldsymbol{d}_{jk}^T \boldsymbol{W}_j$$

In summary, the embedding vector of a given heterogeneous categorical variable is defined as follows:

$$\boldsymbol{x}_t = \begin{cases} \boldsymbol{d}_j^T \boldsymbol{W}_j & \text{single-valued, high-cardinality} \\ \frac{1}{m_t}\sum_k \boldsymbol{d}_{jk}^T \boldsymbol{W}_j & \text{multi-valued, high-cardinality} \\ \boldsymbol{d}_j^T & \text{single-valued, low-cardinality} \end{cases} \tag{3.1}$$

Given above representation, we now define the input data for the neural network as following:

$$\boldsymbol{t} = [n_1, n_2, \ldots, n_k, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_l]$$

The output of the neural network $\hat{y} = f_{model}(\boldsymbol{t})$ is calculated by back-propagating the training loss to both the hidden layer and the embedding layer. Then, the embedding function as well as the parameters for the entire network are updated accordingly.

### 3.3.2 Comparing the Performance of DRG-based Model against the department-based Model

The difference between the treatment scoring model as proposed by Lee et al.[47] and our method roots from data used for training and inference. Lee et al.[47] suggests grouping data by department for training and inference. In contrast, we group the

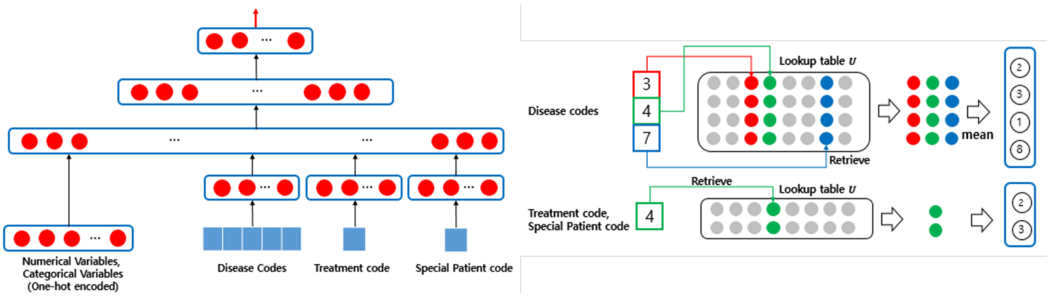


Figure 3.3: The structure of treatment classification model

input data by DRG codes in attempts to reflect the homogeneity of data, while retaining the robustness to seasonality. In this section, we suggest ways to compare out DRG-based model against the department-based model.

Denote the medical treatment set for a given department A for training as $X_A^{trn} = \{\boldsymbol{t}_{A1}, \boldsymbol{t}_{A2}, \dots, \boldsymbol{t}_{An_A}\}$, and the trained model from $X_A^{trn}$ is represented as $f_A$. Moreover, let the medical treatment set with DRG code $k$, for training, be denoted $X_k^{trn} = \{\boldsymbol{t}_{k1}, \boldsymbol{t}_{k2}, \dots, \boldsymbol{t}_{kn_k}\}$, and the trained model from $X_k^{trn}$, as $f_k$. The treatment set for department A for inference is represented as $X_A^{inf} = \{\boldsymbol{t}'_{A1}, \boldsymbol{t}'_{A2}, \dots, \boldsymbol{t}'_{Am_A}\}$, while treatments in the inference set with a DRG code $k$ denoted as $X_k^{inf} = \{\boldsymbol{t}'_{k1}, \boldsymbol{t}'_{k2}, \dots, \boldsymbol{t}'_{km_k}\}$. Now, suppose there appears DRG codes $a, b, \dots, k$ in the given department A. Then, we can train $f_a, f_b, \dots, f_k$ by exploiting $X_a^{trn}, X_b^{trn}, \dots, X_k^{trn}$. Now, for treatments that are prescribed in the department A with DRG code $i$, we represent them, with $X_{Ai}^{inf}$. It can easily be seen that the treatment set prescribed by department A is the union of $X_{Ai}^{inf}$. Mathematically,

$$X_A^{inf} = \bigcup_i X_{Ai}^{inf} = \bigcup_i \{\boldsymbol{t} | \boldsymbol{t} \in X_A^{inf}, \boldsymbol{t} \in X_i^{inf}\}$$

We denote the classification result of $X_A^{inf}$ through the model $f_A$ is denoted as $\hat{Y}_{DEP} = f_A(X_A^{inf}) = \{f_A(\boldsymbol{t})|\boldsymbol{t} \in X_A^{inf}\}$. At the same time, the classification result of $X_{Ai}^{inf}$ via the model $f_i$ is $\hat{Y}_{Ai} = f_i(X_{Ai}^{inf}) = \{f_i(\boldsymbol{t})|\boldsymbol{t} \in X_{Ai}^{inf}\}$. We concatenate $\hat{Y}_{Aa}, \hat{Y}_{Ab}, \ldots, \hat{Y}_{Ak}$ and denote the resulting representation as $\hat{Y}_{DRG}$. Finally, we compare $\hat{Y}_{DEP}, \hat{Y}_{DRG}$, against the true label in order to evaluate the two models' performance. From now on, we define department-based model as the model for calculating $\hat{Y}_{DEP}$ and refer to it as the DEP model. Similarly, we define the DRG-based model as the model for calculating $\hat{Y}_{DRG}$ and refer to it as the DRG model. Figure 3.4 illustrates the entire process.

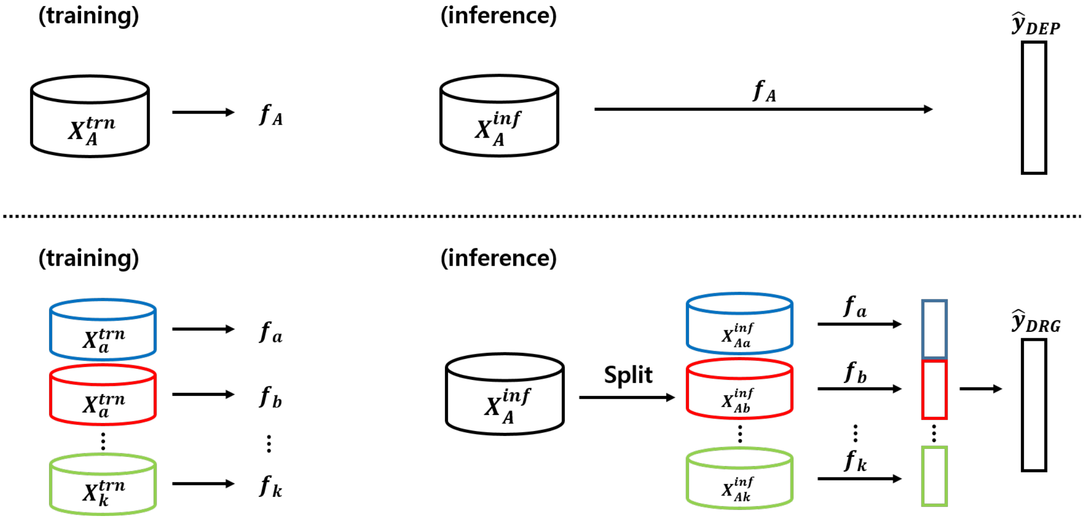


Figure 3.4: Comparison between DEP model and DRG model

## 3.4 Experiments

We evaluate our model on real data which were submitted to HIRA in 2017. We report the performance of our model, with Lee et al.'s [47] as the baseline. The previous method is a scoring model trained with data grouped by department. In subsection 3.4.1, we provide a detailed description of the data as well as the preprocessing steps. We elaborate on performance evaluation metrics in 3.4.2. Finally, training details are presented in 3.4.3.

### 3.4.1 Data Description and Preprocessing

As described in subsection 2.4.1, we worked with several databases that were separately stored within the HIRA data warehouse. Each database stores important features about insurance claims such as claim information, diseases diagnosed and treatments assigned, and the review entailments by the agency. We list the detailed description of each database is presented in Table 2.2. We extracted claim records filed in 2017 that were manually reviewed. [111] During the process, we consulted the on-site field experts and included variables advised as significant by them.

Table 3.1: Data statistics

| Department | Number of Representative DRG codes | Number of claims | Number of treatments | Overtreatment ratio |
|---|---|---|---|---|
| A | 15 | 316,761 | 22,410,573 | 2.13% |
| B | 6 | 45,000 | 2,898,893 | 2.00% |
| C | 2 | 296,238 | 23,331,836 | 5.58% |
| D | 3 | 113,587 | 6,922,504 | 0.67% |
| E | 4 | 169,374 | 7,550,371 | 1.61% |

Following upon the compilation of data, we grouped the resulting claim records by department, as well as the DRG 3-digit code. The reason why we use DRG 3-digit code, instead of the RDRG which is of 6-digit, is because RDRG codes are of too fine granularity. If data is grouped by RDRG codes, then a handful of classes will be empty since grouping is too specific, hence insufficient for learning. As a result, we resorted to DRG 3-digit codes for grouping.

The final complication towards which we should carefully approach during analysis is that more than one DRG 3-digit code may appear even after we group claims by department. For example, given our dataset, approximately two hundred 3-digit DRG codes are observed from the claims filed for the department of internal medicine in 2017. However, the kick here is that the number of treatments for each DRG code follows a long-tail distribution. In other words, a large number of claims cases are associated with only a handful of “important DRG codes”, while few claim reports occur for most of the rest of the DRG codes that are “relatively less important”. Ideally, one would like to models for all the 3-digit DRG codes uniformly across the training phase and compare the performance as illustrated in section 3.3. However, it is impossible, since there are not enough treatment cases with 3-digit DRG codes observed to train the models. Given the restriction on the observed 3-digit DRG codes, we select DRG codes that make up the majority in each department and train the model. Then, we make DRG models using the data that corresponds to the DRG codes. Also, make DEP models using the data corresponds to the data filed from the department and having such DRG codes. Then, we compare the DEP models and DRG models as we already presented in subsection 3.3.2.

The resulting data is processed further following the two important preprocessing schemes: grouping treatments, and integrating treatment codes. First, we cate-

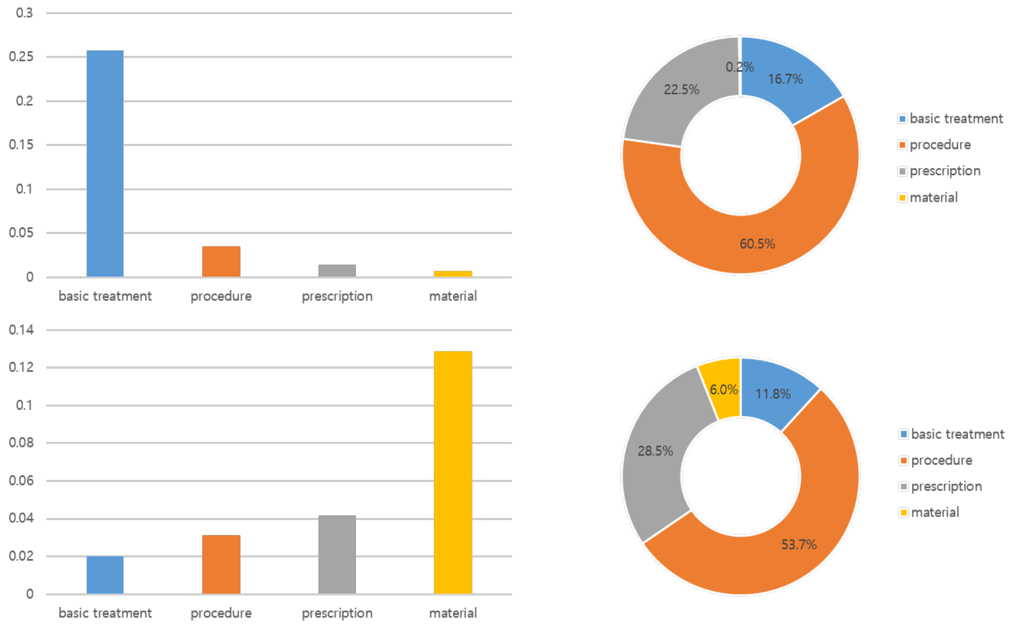
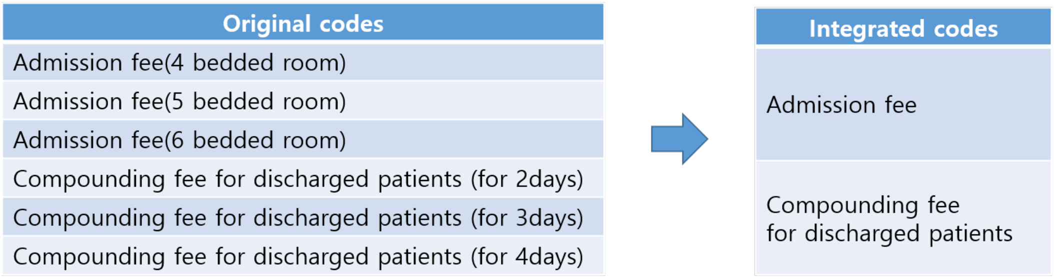


Figure 3.5: The abuse ratio and the distribution of the treatment of two DRG codes

gorize the treatments into four separate groups. The logic for such a process is as follows: suppose that there exists a patient who has received a spine surgery. We assume that the treatments prescribed and practiced for the patient may be compartmented into four distinct categories: the basic treatment, medical procedure, the prescription, and the recovery materials. The basic treatment category includes simple, potentially recurrent medical practices such as admission, consultation, nursing, or providing meals. The medical procedure group entails what practitioners actually conducted on a patient, such as X-rays, MRI examinations, or an operation. The prescription category groups the details of drugs prescribed by the practitioners, such as, for example, the nonsteroidals anti-inflammatory drug. Finally, the recovery materials categorizes all materials needed to recover. A major example of the recovery materials include those for orthosis.

The distribution of DRG codes in data, as well as the class groups, after categorization as described above is quite unbalanced. We take an example and illustrate such a case of class imbalance in Figure 3.5. Figure 3.5 illustrates the class ratio and the distribution of each treatment group of the data with the DRG code. The upper figures are the ratio and the distribution of the data with DRG code of A, while the lower figures are those of the data with DRG code of B. Here, DRG code A corresponds to the appropriate medical DRG codes, while code B corresponds to the surgical DRG codes. It can be easily seen that these two measures behave quite differently from each other. For DRG code A, there are few treatments related to the recovery material group. On the other hand, about 25% of treatments in the basic treatment group are considered to be overtreatment. However, the picture changes completely with DRG code B. Approximately 6.0% of all treatments appear in the recovery material group. At the same time, only 2.5% of the treatments in the basic treatment group are considered to be overtreatment. On the contrary, about 12% of treatments in the recovery material group are considered to be overtreatment. As seen from the above observations, categorizing treatments into more homogeneous groups may lead to more insightful analysis.

Nevertheless, we agglomerated some of the divisions of treatments showing similar characteristics into a single category. For example, Figure 3.6 shows that the category values are too finely grained. HIRA's claim filing process requires for the associated category value to be exactly identified and entered. However, our proposed model does not require such fine granularity in terms of the categories, and it suffices to agglomerate some of the finer categories if their medical implications are similar.



| Original codes |
|---|
| Admission fee(4 bedded room) |
| Admission fee(5 bedded room) |
| Admission fee(6 bedded room) |
| Compounding fee for discharged patients (for 2days) |
| Compounding fee for discharged patients (for 3days) |
| Compounding fee for discharged patients (for 4days) |

| Integrated codes |
|---|
| Admission fee |
| Compounding fee<br>for discharged patients |

Figure 3.6: An example of unifying categories with similar meaning

### 3.4.2 Performance Measures

We compare the performance of our DRG model against the DEP model when the patient distribution changes from the training set to the test set. First, in the case of comparing models with the classification performance of treatments, we follow the process described in subsection 3.3.2.

Before we elaborate on ways to measure the proposed model's performance on classifying different claims filed, we need to draw a clean line between the normal claim and the abnormal claim. In this study, we define an abnormal claim to include more than a single overtreatment assigned or practiced as part of the claim. When we classify claims, not treatments, the problem of class imbalance aggrevates. The ratio of overtreatments to all treatments amounts only to from 0.6% to 5%, while the ratio of abnormal claims among all claims is about from 15% to 40%. In this case, we train several classification models in order to fully utilize claim-level information by employing the decision tree (DT), random forest (RF), neural network (NN), and logistic regression (LR) models.

### 3.4.3 Experimental Settings

The treatment-classification model comprise of a neural network with embedding layers accounting for the categorical variables with high cardinality. ReLU [64] is used as the activation function for the hidden layers to reflect non-linearity when drawing the decision boundary. We employ Adam [39] optimizer with the initial learning rate set at 0.0002. In order to prevent overfitting, we use dropout [85] and early stopping [74] techniques. For categorical variables with high cardinality, we tried different embedding dimensions corresponding to their cardinality. Since the problem of class imbalance eminent, we over-sample data points from the minority class for every batch.

We tested our model with a variety of hyper-parameter settings for each department and DRG code. In each case, we selected a model with the largest area under the precision-recall curve (AUPRC) reported during the validation phase [21]. Then, we selected the best threshold value with the best f1-score. Our selection process is illustrated in Figure 3.7. Pytorch package [70] was used for training the treatment-level information, while scikit-learn package [71] was employed to train to classify the claim-level information.

## 3.5 Results

### 3.5.1 Overtreatment Detection

We report the performance of the treatment classification in Table 3.2. In every case, the claim distribution is simlar between the training set and the test set 1. For the

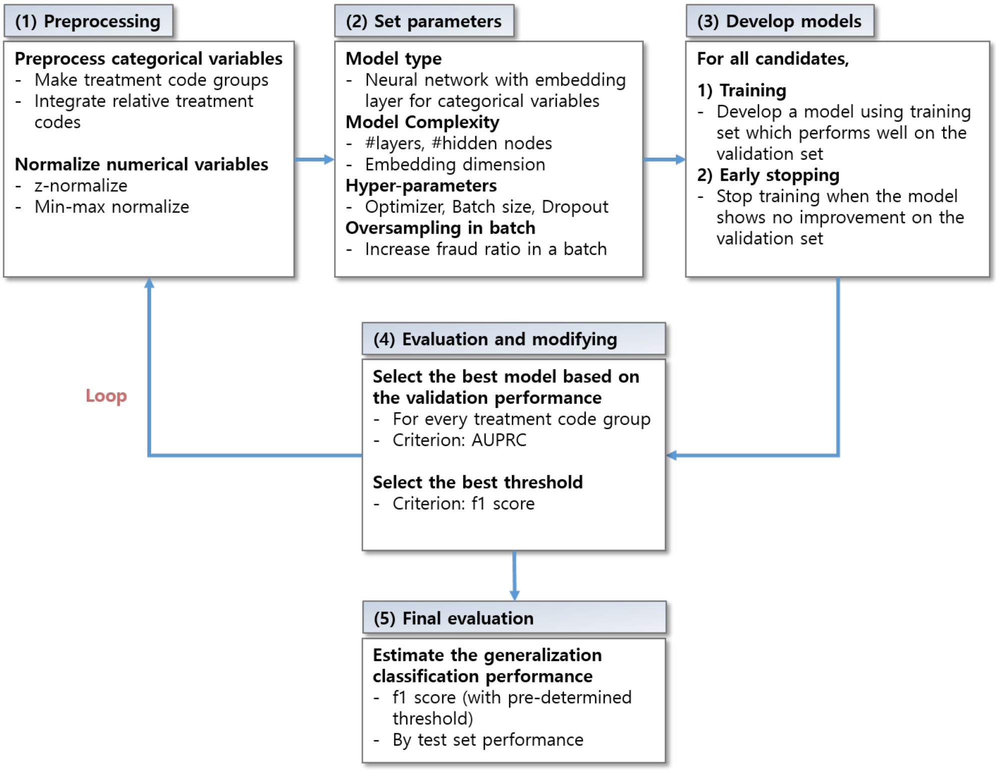

(1) Preprocessing
Preprocess categorical variables
- Make treatment code groups
- Integrate relative treatment codes
Normalize numerical variables
- z-normalize
- Min-max normalize
(2) Set parameters
Model type
- Neural network with embedding layer for categorical variables
Model Complexity
- #layers, #hidden nodes
- Embedding dimension
Hyper-parameters
- Optimizer, Batch size, Dropout
Oversampling in batch
- Increase fraud ratio in a batch
(3) Develop models
For all candidates,
1) Training
- Develop a model using training set which performs well on the validation set
2) Early stopping
- Stop training when the model shows no improvement on the validation set
(4) Evaluation and modifying
Select the best model based on the validation performance
- For every treatment code group
- Criterion: AUPRC
Select the best threshold
- Criterion: f1 score
Loop
(5) Final evaluation
Estimate the generalization classification performance
- f1 score (with pre-determined threshold)
- By test set performance

Figure 3.7: Training and selecting the best abused treatment detection model

departments C and E, the distribution is similar between the test set 1 and the test set 2. In other cases, there is a difference in the distribution between test sets.

As for test set 1, the DEP model performs slightly better than the DRG model in most cases. It can be seen that the proposed model potentially learns different types of patients better. However, for the test set 2, different results are observed. In most cases, the DRG model performs better as compared to the DEP model. Moreover, the decrease in the performance of the DEP model is quite dramatic, while the DRG model shows relatively more stable performance. So, when the distribution for most prominent patient type is shifted, the DEP model fails to perform well. However, as for the DRG models, every model trains the treatment pattern according to the each patient type. Hence, even if the patient distribution changes, the degradation in performance is not so severe. Altogether, we conclude that the DRG model is more robust to the change in the distribution of the patient type as compared to the DEP model when classifying the treatments.

Table 3.2: Performance of the overtreatment detection

| Department | Model | Accuracy | | Precision | | Recall | | F1 | |
|---|---|---|---|---|---|---|---|---|---|
| | | Test1 | Test2 | Test1 | Test2 | Test1 | Test 2 | Test1 | Test2 |
| A | *DEP* | 0.9473 | 0.9565 | 0.2132 | 0.2027 | 0.4305 | 0.2809 | **0.2852** | 0.2355 |
| | *DRG* | 0.9485 | 0.9560 | 0.2090 | 0.2197 | 0.3979 | 0.3318 | 0.2741 | **0.2644** |
| B | *DEP* | 0.9678 | 0.9769 | 0.2907 | 0.3545 | 0.4227 | 0.2501 | **0.3445** | 0.2933 |
| | *DRG* | 0.9642 | 0.9703 | 0.2575 | 0.2739 | 0.4178 | 0.3303 | 0.3186 | **0.2994** |
| C | *DEP* | 0.9423 | 0.9373 | 0.5247 | 0.5017 | 0.6382 | 0.6259 | **0.5759** | 0.5569 |
| | *DRG* | 0.9419 | 0.9403 | 0.5221 | 0.5210 | 0.6378 | 0.6472 | 0.5742 | **0.5773** |
| D | *DEP* | 0.9790 | 0.9745 | 0.1333 | 0.0347 | 0.4025 | 0.1285 | 0.2003 | 0.0546 |
| | *DRG* | 0.9901 | 0.9922 | 0.2768 | 0.2301 | 0.3227 | 0.1479 | **0.2980** | **0.1801** |
| E | *DEP* | 0.9807 | 0.9809 | 0.3340 | 0.3927 | 0.4416 | 0.3884 | **0.3803** | **0.3905** |
| | *DRG* | 0.9780 | 0.9780 | 0.2952 | 0.3381 | 0.4631 | 0.4122 | 0.3606 | 0.3715 |

### 3.5.2 Abnormal Claim Detection

Table 3.3 reports the performances of DEP models, DRG models, and the models that utilize claim-level variables. Above all, we can see that the DEP models and the DRG models exploiting the treatment-level variables perform better than the other models. It implies that the models with the treatment-level variables may perform better than the models with the claim-level variables when classifying claims.

Also, the degradation in the performance of the DEP model from the test set1 to the test set 2 is clearly apparent; yet, the decrease in performance for the DRG model is not as large as the DEP model. It may suggest that the DRG model is more robust to the changes in the distribution of the patient type than the DEP model when classifying the claims.

Table 3.3: Performance of the abnormal claim detection

| Department | Model | Accuracy | | Precision | | Recall | | F1 | |
|---|---|---|---|---|---|---|---|---|---|
| | | Test1 | Test2 | Test1 | Test2 | Test1 | Test 2 | Test1 | Test2 |
| A | *DEP* | 0.5989 | 0.6156 | 0.5013 | 0.5193 | 0.8593 | 0.7486 | **0.6332** | 0.6132 |
| | *DRG* | 0.5969 | 0.6186 | 0.4998 | 0.5205 | 0.8627 | 0.7988 | 0.6329 | **0.6306** |
| | *LR* | 0.4976 | 0.5128 | 0.4282 | 0.4402 | 0.9465 | 0.9473 | 0.5896 | 0.6011 |
| | *NN* | 0.4973 | 0.5131 | 0.4286 | 0.4408 | 0.9555 | 0.9558 | 0.5918 | 0.6034 |
| | *DT* | 0.5763 | 0.5747 | 0.4396 | 0.4451 | 0.4050 | 0.3960 | 0.4213 | 0.4191 |
| | *RF* | 0.4771 | 0.4893 | 0.4165 | 0.4267 | 0.9266 | 0.9261 | 0.5747 | 0.5843 |
| B | *DEP* | 0.6568 | 0.6709 | 0.5213 | 0.5942 | 0.7095 | 0.4940 | **0.6010** | 0.5395 |
| | *DRG* | 0.6179 | 0.6295 | 0.4847 | 0.5200 | 0.7750 | 0.6643 | 0.5964 | **0.5832** |
| | *LR* | 0.4558 | 0.4688 | 0.3967 | 0.4171 | 0.9277 | 0.9146 | 0.5558 | 0.5729 |
| | *NN* | 0.4462 | 0.4592 | 0.3939 | 0.4134 | 0.9451 | 0.9266 | 0.5561 | 0.5717 |
| | *DT* | 0.5761 | 0.5744 | 0.4178 | 0.4463 | 0.3941 | 0.3837 | 0.4056 | 0.4126 |
| | *RF* | 0.4469 | 0.4555 | 0.3932 | 0.4110 | 0.9339 | 0.9177 | 0.5534 | 0.5677 |
| C | *DEP* | 0.7085 | 0.7129 | 0.6951 | 0.7198 | 0.9207 | 0.9043 | **0.7922** | 0.8016 |
| | *DRG* | 0.7085 | 0.7200 | 0.6984 | 0.7266 | 0.9097 | 0.9035 | 0.7902 | **0.8054** |
| | *LR* | 0.5776 | 0.6069 | 0.5762 | 0.6069 | 0.9941 | 0.9934 | 0.7296 | 0.7535 |
| | *NN* | 0.5827 | 0.6126 | 0.5792 | 0.6103 | 0.9942 | 0.9937 | 0.7319 | 0.7562 |
| | *DT* | 0.6037 | 0.5968 | 0.6624 | 0.6845 | 0.6293 | 0.6181 | 0.6454 | 0.6496 |
| | *RF* | 0.6007 | 0.6293 | 0.5920 | 0.6237 | 0.9754 | 0.9755 | 0.7368 | 0.7609 |
| D | *DEP* | 0.5621 | 0.4753 | 0.2337 | 0.1918 | 0.7017 | 0.6709 | 0.3506 | 0.2983 |
| | *DRG* | 0.8038 | 0.8033 | 0.4347 | 0.3749 | 0.5480 | 0.2741 | **0.4848** | 0.3167 |
| | *LR* | 0.6381 | 0.6119 | 0.2416 | 0.2643 | 0.6180 | 0.6116 | 0.3474 | **0.3691** |
| | *NN* | 0.8443 | 0.8143 | 0.6667 | 0.3846 | 0.0020 | 0.0012 | 0.0041 | 0.0024 |
| | *DT* | 0.7619 | 0.7346 | 0.2431 | 0.2622 | 0.2497 | 0.2372 | 0.2464 | 0.2491 |
| | *RF* | 0.5295 | 0.4977 | 0.2115 | 0.2264 | 0.7398 | 0.7062 | 0.3289 | 0.3429 |
| E | *DEP* | 0.7119 | 0.6893 | 0.5293 | 0.5311 | 0.6970 | 0.6076 | **0.6017** | 0.5668 |
| | *DRG* | 0.6999 | 0.6868 | 0.5138 | 0.5266 | 0.7188 | 0.6308 | 0.5992 | **0.5740** |
| | *LR* | 0.6068 | 0.6089 | 0.3689 | 0.3914 | 0.8622 | 0.8471 | 0.5167 | 0.5354 |
| | *NN* | 0.6008 | 0.6015 | 0.3682 | 0.3894 | 0.8907 | 0.8768 | 0.5210 | 0.5393 |
| | *DT* | 0.6923 | 0.6824 | 0.3676 | 0.3946 | 0.3643 | 0.3628 | 0.3659 | 0.3780 |
| | *RF* | 0.5641 | 0.5627 | 0.3468 | 0.3675 | 0.8916 | 0.8935 | 0.4993 | 0.5208 |

## 3.6 Summary

The distribution of health insurance claims shifts from time to time due to seasonality of several diseases. Nevertheless, there are few abuse detection models which effectively account for seasonality. Most studies employ coarsely grained derived variables, defined at the provider or claim-level, for example, which makes it even more difficult to address the seasonality issues when modelling abuse detection algorithms.

In the previous chapter, we proposed an abusive provider detection model using treatment-level information. The proposed model detects abusive providers for each given department. The underlying assumption of the proposed abusive provider detection model is that claims are similarly distributed in the training and the test sets. This assumption may not hold true for some departments. If we ignore this difference in modeling, the performance will be decreased.

In order to tackle seasonality issues, we implement an abuse detection model which incorporates treatment classification to detect abuse cases for each DRG code. DRG is a type of the patient classification system (PCS), which classifies patients into groups based on clinical features and the consumption pattern of medical resources. We observe that claims with the same DRG code show similarity regardless of the timing of the filing. Instead of running a single model separately for each department, we propose to a model embodying multiple structures specific to DRG codes selected as important for each given department. We also run the single model for each department and compare the results with our proposed model. Experiment results show our proposed model performs well across different time windows, while the department-wise single models show degradation in performance.

This paper contributes to the existing literature by building the abuse detection model which effectively accounts for seasonality in health insurance claims. Moreover, we provide ground evidence for DRG, an ontology originally designed to categorize patients, to be used in the claim review process.

# Chapter 4

# Detection of overtreatment with graph embedding of disease-treatment pair

## 4.1 Background

Practitioners can prescribe a wild range of different treatments for the same patient. Moreover, there exist myriads of drugs that share the same efficacy. Yet, practitioners have a tendency to stick to their preferred choice of the drug and prescribe it to their patients, even though other options are available. The product may be selected based on the practitioners' clinical experience or personal preference. This same affinity towards specific choices can be observed not only from drug prescription but also from practicing medical procedures.

Suppose there are two practitioners who prescribe different drugs, which actually have similar medical efficacy, to the same patient. Two separate claims will be filed for each practice. Now, when the reviewers examine these claims, based on their expertise, it can easily be determined that both cases are normal since both prescriptions are appropriate responses to patient's disease. The machine, however, will have to establish such relational knowledge from scratch, and it will have to learn it from data. However, previously suggested models are not designed to efficiently deal with the complex relationships between the disease and the treatments.

In the previous chapters, embedding vectors for both diseases and treatments are learned simultaneously. Hence, these embedding vectors are in separate spaces. In order to add the relationship between disease and treatment, we simply concatenate the embedding vectors additionally feed to the model. However, it is not sufficient of an approach to include complex disease-treatment relationship.

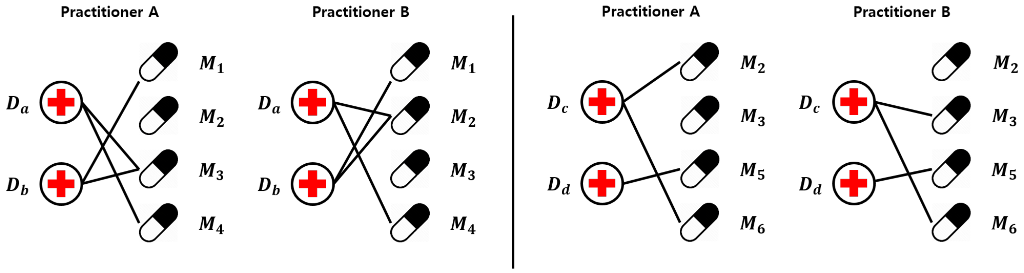


Figure 4.1: An example of different prescription from different practitioners to the same patient

Let us illustrate the reasoning behind this assertion by taking a toy example. Suppose there is a claim in the test set with the same diseases as found in some of the claims in the training set. Suppose, however, treatments prescribed in the test claim are different from those prescribed in the training claims, even though the patients in these claims suffered from similar diseases. A naïve model will classify the treatment prescribed in the test set at random, because it is a practice pattern unseen during the training. The naive model does not know that the diseases in both train and test claims are similar to each other. It won't be able to learn that, even though the prescribed treatments differ in the train claim and the test claim, the medical efficacy of the treatments are actually very similar. One the contrary, if the correct disease-treatment relationship can be modeled before the training, then the following abuse detection model would certainly perform better.

In this chapter, we propose an overtreatment detection model which considers the intricate disease-treatment relationships in prior to training. The proposed method consists of two stages. During the first stage, the disease-treatment network is constructed from the claims data. During the second stage, the model is trained to learn vector representations of entities from the disease-treatment network using node embedding methods. With the trained embedding vectors, we predict link formation between treatment and diseases in the claim in order to determine whether the treatment listed in the given claim is unnecessary to the subject patient. We test employing different network embedding models and suggest strategies to choose the most appropriate method. Our selection metric is the average performance on link prediction between the disease and the treatment.

The rest of the chapter is organized as follows. In section 4.2, we review the literature on graph embedding methods and the applications of the graph embedding method in biomedical data. Also, we introduce several studies about medical concepts embedding. Section 4.3 provides detailed descriptions of the proposed model. In section 4.4, we elaborate on experiment settings. We also describe the data in this section. Section 4.5 reports the experiment results. Finally, section 4.6 concludes the paper.

## 4.2 Literature review

In this section, we explained some state-of-the-art graph embedding network methods and their application to biomedical data. In subsection 4.2.1, we briefly introduced some graph embedding methods. In subsection 4.2.2, we reviewed about applying graph embedding methods in biomedical data. Finally, We described some methods related to medical concept embedding in section 4.2.3.

### 4.2.1 Graph embedding methods

The graph embedding methods can be divided into four categories: matrix factorization(MF)-based methods, random walk-based methods, deep learning-based methods, and other methods. In this subsection, we briefly reviewed each category and corresponding methods.

**MF-based methods**

Originally, the matrix factorization method has been widely adopted for dimension reduction of the data matrix. The data matrix is factorized into lower-dimensional matrices while preserving the manifold structure. The MF-based graph embedding method is factorizing matrices, which represent graph properties, to obtain node embedding vectors in lower dimension space. There are several graph embedding methods that utilize matrix factorization methods such as locally linear embedding(LLE) [76], Laplacian eigenmaps(LE) [7], Graph Factorization(GF) [2], GraRep [9], and HOPE [69].

In LLE, find k-nearest neighbors(k-NN) of each data and make an adjacency matrix based on the k-NN result. Then, factorize the matrix using the matrix factorization method such as Singular Value Decomposition(SVD). While LLE [76] uses the constructed matrix itself, LE [7] factorizes graph Laplacian Eigenmaps to preserve pairwise node similarities. It converts finding embedding vector problem to generalized eigenvector problem. GF [2] directly factorize the proximity matrix of a graph under each edge is already existed. These methods aim to preserve 1st-order proximity. However, many networks have important features in high-order proximities.

GraRep and HOPE are two important methods that preserve high-order proximities. In GraRep [9] method, authors capture network the local and global structure by generating multiple $k$-step embedding vectors by factorizing multiple $k$-step transition probability matrices, and concatenate those vectors. HOPE [69] is used to get embedding vectors of the directed graph which has asymmetric transitivity. The basic idea of HOPE is that a node should have two different embedding vectors because each node can be used as a target node. It defines some important high-order proximity measures such as Katz index [37], and get embedding vectors that preserve such measures. SVD is used to factorize the matrices in both methods.

**Random walk-based methods**

Random walk is a stochastic process with random variables $W^1_{v_i}$,$W^2_{v_i}$,...,$W^k_{v_i}$ such that every value is randomly chosen from the neighbors of previous value. In other words, if $W^j_{v_i} = v_j$, then $W^j_{v_i}$ must be randomly chosen from $N(v_j)$, which means the neighbors of node $v_j$. In short, a random walk in a network is a node sequence in which every node is connected to the previous node. It is commonly used to capture structural relationship between nodes of the network. Perozzi et al. [72] found that the distribution of vertices appearing in short random walks is similar to the distribution of words appearing in sentences under certain circumstances. Inspired by this observation, they suggested a method named DeepWalk, which utilizes SkipGram [61] model in random walks to learn the embedding vector of each node. Also, the hierarchical softmax method was used to train SkipGram model ([62], [63]). After this study, there have been several papers that utilize the word embedding model in

NLP to random walks. Grover and Leskovec [29] suggested node2vec model that uses the biased random walk rather than unbiased random walk in DeepWalk. They used the biased random walk to preserve the local structure and the global structure by using breadth-first searching and depth-first searching in generating random walks. With these random walks, they trained SkipGram model with negative sampling. Perozzi et al. [73] proposed Walklets, which is another extension of DeepWalk. They modified a strategy of generating random walk to skipping some nodes each random walk. By this strategy, they made it possible to generate random walks that contain multiple $k$-steps proximities.

Diffusion component analysis (DCA) [13] is another random walk-based embedding method, but quite different from previous methods. While previous methods are node embedding methods utilizing SkipGram model, DCA calculates the diffusion state that is defined as the probability distribution in stationary state with random walk with restart (RWR) strategy. This strategy captures both local and global structural property. Also, it makes possible to overcome the noise and sparsity of the network, so that this method can be used in the biological network.

While these methods were concerned only about proximities, struc2vec [75] is a graph embedding method that preserves structural identity. The authors of struc2vec explain the structural identity as *a concept a symmetry in which network nodes are identified according to the network structure and their relationship to other nodes* [75]. In other words, a node pair having structural identity means both nodes perform similar roles in the network. Firstly, define the structural similarity and construct a multilayer weighted network where all nodes exist in every layer. Then, generate the context for each node by using biased random walks with the multilayer network.

Then, train a SkipGram model with the hierarchical softmax method to learn node embedding vector for each node with random walks.

These methods are for the homogeneous networks, which refer to networks with a single type of nodes and edges. However, there are much more networks which are not homogeneous such as author-paper-venue, customer-products-seller network. These networks are called heterogeneous networks which include different types of nodes and edges. There are several studies about embedding methods for these networks, such as metapath2vec [23] and Heterogeneous Information Network Embedding (HINE) [35]. Here, both methods are random walk-based method. Except for generating random walks method, metapath2vec is quite similar to DeepWalk. They suggest meta-path based random walks for the heterogeneous network, which generates random walks by pre-defined node type sequence. Otherwise, HINE [35] does not utilize SkipGram method in training. The authors first defined two meta-path based proximity measures for a heterogeneous network. Then, train embedding vectors of nodes while preserving those proximities.

**Deep learning-based methods**

Deep learning has been achieved success in various domains. Deep learning-based embedding methods are the embedding methods that utilize some deep learning architectures. SDNE [92] is a kind of node embedding model which utilizes the deep auto-encoder to proximity matrix of the network to map it to nonlinear latent space while preserving the network structure. By using the auto-encoder, the embedding vector preserves the second-order proximity. It makes 1st order proximity also be preserved by applying Laplacian eigenvector proximity measure to embedding vectors. DNGR [10] is another model that utilizes auto-encoder structure The authors chose the stacked denoising auto-encoder structure to find non-linear embedding vectors in low dimensional space and robust to the noise of the network. Graph convolutional network (GCN) [41] and variational graph auto-encoder (GAE) [40] are also important deep learning-based model. Both of them use the convolutional neural network (CNN) in network data which achieves great success in the computer vision domain. GCN applies the convolutional operation to network data by using the proximity matrix and feature matrix of the network. GAE is a kind of auto-encoder that uses GCN encoder and the simple inner product decoder.

**Other methods**

There are several important methods that are not included in any category. Multidimensional scaling(MDS) [33] learns embedding vectors by preserving the distance of all node pairs in the embedding space. However, it does not consider different relationships might have different importance. Isomap [88] overcome this shortage by constructing k-NN network and learn embedding vectors while preserving the distance between a node and its k-NNs. LINE [86] is a node embedding method that preserves the first-order and second-order proximity. The authors suggested preserving 1st order proximity by minimizing the distance between the empirical distribution of nodes in the original graph and the distribution from embedding space. Also, they suggest minimizing the distance between the empirical conditional distribution of ‘context’ node $v_j$ given a single node $v_i$ and the conditional distribution of them in the embedding space.

Table 4.1: graph embedding methods

| Category | Algorithm | Method |
|---|---|---|
| Matrix factorization | LLE [76] | matrix factorization (e.g. SVD) |
| | LE [7] | matrix factorization (e.g. eigen-decomposition) |
| | GF [2] | matrix factorization (e.g. SVD) |
| | GraRep [9] | matrix factorization (e.g. SVD) |
| | HOPE [69] | matrix factorization (e.g. SVD) |
| Random walk | DeepWalk [72] | skip-gram with random walk |
| | node2vec [29] | skip-gram with random walk |
| | Walklets [73] | skip-gram with random walk |
| | DCA [13] | stationary distribution with random walk with restart strategy |
| | struc2vec [75] | skip-gram with random walk |
| | metapath2vec [23] | skip-gram with meta-path based random walk |
| | HINE [35] | proximity preserving model with meta-path based random walks |
| Deep learning | SDNE [92] | Autoencoder with proximity matrix |
| | DNGR [10] | Denoising autoencoder with PPMI matrix |
| | GCN [41] | CNN model with adjacency matrix and feature matrix |
| | GAE [40] | Autoencoder with GCN encoder and simple inner product decoder |
| Others | MDS [33] | Preserving Euclidean distances of all node pairs |
| | Isomap [88] | Preserving Euclidean distances of each node and its $k$-nearest neighbors |
| | LINE [86] | Preserving 1st-order and 2nd-order proximity |

### 4.2.2 Application of graph embedding methods to biomedical data analysis

The network embedding method is applied mainly in three topics: pharmaceutical data analysis, multi-omics data analysis, and clinical data analysis. In this subsection, we explained each category and review several studies.

**Pharmaceutical data analysis**

The usage of graph embedding or graph analysis in pharmaceutical data can be categorized as three important issues: drug-target interaction (DTI) prediction, drug-drug interaction (DDI) prediction, and drug-disease association (DDA) prediction. DTI prediction means predicting the interactions between drugs (chemical compound) and target (protein). DDI prediction is to predict the result of drug co-prescription. DDA prediction means predicting the clinical result when a patient, who has a specific disease, takes a specific drug.

Previously, DTI prediction was mainly performed by constructing proximity matrices and factorize them by matrix factorization methods. Yamanashi et al. [102] proposed a method of predicting unknown DTI by using known DTI data, chemical data, and genomic data. They construct a known drug-target bipartite network by DTI data and factorize the similarity matrix by eigenvalue decomposition. Next, train models that represent the correlation between embedding space and chemical/genomic space. Then, the unknown DTI can be inferred by the model. Cobanoglu et al. [18] proposed a method that predicting DTI by using the collaborative filtering method only with known DTI data, not any external data. They applied the probabilistic matrix factorization method to the known DTI network to get embedding vector of each node in drug-protein and predict unknown DTI by active learning with learned embedding vectors. Ezzat et al. [25] suggested a method that predicts DTI by the graph embedding method and ensemble learning. They conducted a feature sub-spacing to inject diversity for classifier ensemble and tried three different dimension reduction methods: SVD, Partial Least Squares(PLS), and LE. Then, train homogeneous base learners with the resulting vectors and predict with each model's score. Also, there is another method that uses the k-NN method and graph regularization matrix factorization method to predict unknown DTI [26].

While MF-based methods were used in previous studies, random-walk based methods are also commonly used in DTI prediction. Luo et al. [54] developed a model named DTINet to predict DTIs from a heterogeneous network that is constructed by integrating drug-related information. They used extended DCA to learn embedding vectors for each node in the heterogeneous network. Then find the best projection from drug space to target space by finding mapped feature vectors of drugs are similar to the known interacting target. Then, infer new interactions of a drug by ranking the target candidates and projected feature vector of the drug. Zong et al. [110] proposed a similarity-based DTI prediction method by constructing a drug-target-disease tripartite network. After construction, train embedding vectors for each node to predict the drug-target association. They utilized DeepWalk method to learn embedding vectors. Alshahrani et al. [3] proposed another method that integrates external information to construct a heterogeneous network. They integrated gene ontology(GO), protein-protein interactions(PPIs), DTIs, gene-disease interactions, drug side effects, and disease-phenotype information to construct the network. They utilized a modified DeepWalk method to learn embedding vectors that captures the structure of the network. Then, they trained the logistic regression model to predict the unknown DTIs.

There were also several studies predicting DDIs. Zhang et al. [107] proposed a method that formulates DDI prediction as a matrix completion problem. Firstly, they integrated multiple external drug-related information and learned embedding vectors. Also, they suggested a method named 'Manifold Regularized Matrix Factorization' (MRMF), which is a kind of MF-based embedding method, to learn embedding vectors. Then, they found similarity factors between node pairs with embedding vectors and known DDI information. Ma et al. [56] proposed a model that calculates similarities between drugs in multi-view. They used GAE to integrate multiple types of drug features and attentive model to make the model adaptive to data. They also used the model to predict unknown DDI. Zitnik et al. [109] proposed a model named 'Decagon', which is aimed to predict DDI, especially polypharmacy side effects. Firstly, they constructed a multimodal graph from PPIs, DTIs, and polypharmacy side effect information. Different types of interactions are labeled by different edge types. The unknown DDIs are predicted by link prediction between drug nodes using modified GAE. The node information is encoded by the GCN based encoder. Then, the decoder takes pairs of embedding vector and scores the edge between them.

Predicting DDAs is also an important issue in pharmaceutical data analysis. Dai et al. [20] first embedded gene-gene interaction network by eigenvalue decomposition and get embedding vectors of drugs and disease with the gene embedding vectors, drug-gene interactions, and disease-gene interactions. Then, factorize the known drug-disease association matrix. Finally, the unknown DDAs can be inferred by the embedding vectors of drugs and diseases, and the matrix factorization result of known drug-disease association matrix. Wang et al. [94] constructed a drug-disease network from free text, especially extracted from papers in PubMed and learn embedding vector with modified LINE. Then, the correlation of the drug-disease pair is calculated by the embedding vectors and find DDA patterns.

**Multi-omics data analysis**

The term 'omics' means a field of study in biology that ends with '-omics', such as genomics. These studies are about researching characteristics of biological molecules such as their structures, functions, or dynamics. The network-based approach is a valuable method in these studies in finding a relationship between entities. Here, we reviewed three important topics that utilize graph embedding methods: proteomics, genomics, and transcriptomics data analysis.

Many studies that apply graph embedding methods in proteomics is focused on assessing and predicting PPIs, or predicting protein functions. Kuchaiev et al. [45] proposed a de-noising PPIs model with MDS-based graph embedding approach to address high false positive and false negative in PPIs. You et al. [104] used isomap to embed the PPI network in low dimensional space. Then, assess and predict the PPIs by comparing embedding vectors of the node pair. Lei et al. [49] proposed a two-step model that assesses and predicts PPIs. First, combine multiple genomic and proteomics information by logistic regression approach to construct a weighted PPI network. Then, get embedding vectors by extended isomap and predict the unknown PPIs. Wang et al. [97] proposed ProsNet, which predicts the PPI by constructing a heterogeneous molecular network and embedding the network in low dimensional space. The heterogeneous molecular network is constructed by including the molecular networks of several species and gene ontology graph. Then, the embedding vectors are calculated by meta-path based extended DCA.

Graph embedding methods are utilized for various purposes in analyzing genomic data. As We already reviewed in the previous subsection, Cho et al. [13] proposed DCA, which is an important graph embedding method, to learn node embedding vectors with RWR strategy. Wang et al. [95] proposed a method named clusDCA, which predicts the gene function. They learned embedding vectors from gene-gene interaction network and GO by DCA. Then, they trained a projection model from gene space to GO space. With projected vectors and embedding vectors in GO space, they predicted the gene function of the gene. There is also another DCA-based model named PACER that aims to pathway identification [96]. The main idea of this method is to construct a heterogeneous network and embedded gene and pathway in a unified space. They used gene expression, drug response-gene expression, PPIs, and pathway information to construct the network. Li et al. [51] proposed a model named SCRL, which aims to learn the representation of a single cell RNA sequence. The basic idea of this model is constructing cell-ContexGene and Gene-ContextGene networks and learning embedding vectors by extended LINE. Zeng et al. [106] constructed a heterogeneous gene-disease network from human genes and other species' genes information. Then, they calculated embedding vectors by factorizing the matrix and predicted the pathogenic human genes.

Transcriptomics is a study of an organism's transcriptome, which is all about RNA transcript. In this field of study, The graph embedding methods are mainly used to identify the miRNA-disease association. Shen et al. [81] developed Collaborative Matrix Factorization for miRNA-Disease Association(CMFRDA) that identifies the miRNA-disease association. They constructed a miRNA-disease bipartite graph and factorized the matrix by the SVD for initialization. Then, they update the factorized matrix until the predefined loss is converged. Li et al. [50] proposed a similarity-based miRNA-disease prediction model. They constructed the miRNA-disease bipartite network and learned similarity by embedding the network with DeepWalk. Then, they infer the miRNA-disease interaction by the distance between embedding vectors of a node pair.

**Clinical data analysis**

There are several papers about analyzing the clinical data, such as medical knowledge graph, electronic health records (EHRs) and electronic medical records (EMRs). Choi et al. [16] suggested learning embedding vectors from three different data sources: medical journals, medical claims, and clinical narratives. Different types of concepts are embedded in a common low-dimensional space. They tried two embedding methods: SkipGram and matrix factorization. Wang et al. [93] suggested a method to recommend appropriate medicine for patients. They constructed heterogeneous network by combining medical knowledge network, patient-medicine network, and patient-disease network. They trained embedding vectors of the network by using translation-based embedding method and LINE. Choi et al. [15] developed a model named GRAM, which aims to learn low-dimensional representation with medical concept ontology. They utilized the attention method to leverage the parent-child relationship of the ontology.

Table 4.2: Applications of graph embedding methods in biomedical data analysis

| Tasks | Authors | Purpose | Embedding method |
|---|---|---|---|
| Pharmaceutical data analysis | Yamanashi et al. [102] | DTI prediction | Matrix factorization |
| | Cobanoglu et al. [18] | DTI prediction | Probabilistic matrix factorization |
| | Zheng et al. [108] | DTI prediction | Matrix factorization |
| | Ezzat et al. [25] | DTI prediction | Matrix factorization |
| | Ezzat et al. [26] | DTI prediction | Matrix factorization (SVD, PLS, LE) |
| | Luo et al. [54] | DTI prediction | DCA |
| | Zong et al. [110] | DTI prediction | DeepWalk |
| | Alshahrani et al. [3] | DTI prediction | Modified DeepWalk |
| | Zhang et al. [107] | DDI prediction | matrix factorization |
| | Ma et al. [56] | DDI prediction | GAE |
| | Zitnik et al. [109] | DDI prediction | modified GAE |
| | Dai et al. [20] | DDA prediction | Eigenvalue decomposition, Matrix factorization |
| | Wang et al. [94] | DDA prediction | modified LINE |
| Multi-omics data analysis | Kuchaiev et al. [45] | Denoising PPI | Extended MDS |

Table 4.2: Applications of graph embedding methods in biomedical data analysis

| Tasks | Authors | Purpose | Embedding method |
|---|---|---|---|
| | You et al. [104] | Assessing PPI, PPI prediction | Isomap |
| | Lei et al. [49] | Assessing PPI, PPI prediction | Extended Isomap |
| | Wang et al. [97] | PPI prediction | Meta-path based extended DCA |
| | Cho et al. [13] | Node embedding in biological network | DCA |
| | Wang et al. [95] | Gene function prediction | Extended DCA |
| | Li et al. [51] | Learn the representation of single cell RNA-seq | Extended LINE |
| | Zeng et al. [106] | Predict pathogenic human genes | Matrix factorization |
| | Wang et al. [96] | Pathway identification | DCA |
| | Shen et al. [81] | Identify potential miRNA-disease association | Matrix factorization |

Table 4.2: Applications of graph embedding methods in biomedical data analysis

| Tasks | Authors | Purpose | Embedding method |
|---|---|---|---|
| | Li et al. [50] | miRNA-disease prediction | DeepWalk |
| Clinical data analysis | Choi et al. [16] | Medical concept embedding | SVD, SkipGram |
| | Choi et al. [15] | Medical concept embedding | GRAM |
| | Wang et al. [93] | Medicine Recommendation | Translation based, LINE |

### 4.2.3 Medical concept embedding methods

In order to apply various machine learning methods in clinical data, the medical concept in the clinical data should be vectorized. There have been researches to embedding medical concept to get embedding vector for various purpose, such as predicting patients' visits.

Choi et al. [14] proposed a medical concept representation method named med2vec from EHR datasets. Here, they define a visit vector $V_t$, and represent it as a binary vector $x_t \in 0, 1^{|C|}$ , where the $i$-th entry is 1 only if $c_i \in V_t$. Then, represent the binary vector in intermediate low dimensional space, and concatenate the vector with demographic information. Embed the concatenated vectors into the final low dimensional space and predict the other binary vectors in a context window. Not only they used inter-visit information, but also they used inter-visit information to preserve code-level information. Another work done by the Choi's team is GRAM [15], which is we already reviewed in the subsection 4.2.2. It utilized the attention model to each node in the medical concept ontology to learn low-dimensional embedding vectors that leverage the parent-child relationship. Song et al. [83] proposed another ontology-based medical concept model named MMORE. The model learns multiple embedding vectors for the ancestors of the leaf nodes and the final embedding vectors are calculated by combining those embedding vectors with an attention mechanism.

Cai et al. [8] proposed an embedding method that considers the temporal information because the scopes medical concept varies greatly in terms of temporal scope. The embedding vectors are calculated from other EMR data codes that are in a certain time window with the attention model. Xiang et al. [101] claimed that the embedding vectors of medical concepts should consider temporal dependency. They tried word2vec, PPMI, and FastText [36] with large EHR datasets to learn embedding vectors of medical concepts to overcome this issue.

## 4.3 Proposed method

This section presents our overtreatment detection model using graph embedding method. Subsection 4.3.1 details the process of the medical information network from the medical treatments found in healthcare insurance claims. More specifically, we present how to compute the edges of the network from the treatment data. Subsection 4.3.2 describe our strategies for choosing the best method for embedding the constructed network in order to carry out the overtreatment detection task. We solve the link prediction problem and compare performances among the select methods.

Finally, we construct the overtreatment detection model by using graph embedding in subsection 4.3.3. We sample negative edges from the constructed network, in particular, and use the embedding vectors of the nodes, trained and chosen as described in subsection 4.3.2, to predict links.

### 4.3.1 Network construction

Our healthcare insurance claim data consists of three parts: (1) basic claim information; (2) disease information, and; (3) treatment information. Basic claim includes claim-wise elements as claim identifiers, general practitioner (GP) information, subject patient profiles, as well as the relevant DRG code. Disease information reports the list of diseases the subject patient has. The treatment information encompass all the details of treatments that the general practitioner has prescribed the patient. Figure 2.5 illustrates an example of a claim typically found in our data set

The toughest challenge in constructing a network from in insurance claim data set is that the casual relationship between the disease and the treatment is not apparent. A claim contains information on the main and sub-diseases diagnosed as well as the type and amount of treatments, yet it still remains in dark exactly for which disease each treatment was prescribed. The absence of exact disease-treatment matching poses as a problem in the following sense: suppose an edge between a disease and a treatment is formed if they appear in the same claim. For example, if diseases A and B are listed together with treatments C and D for an arbitrary claim case c, then an edge will be formed between disease A and treatments B and C. Now, suppose that, in reality, treatment D was prescribed for disease B only and, likewise, treatment C only for disease A. Then, the edge between disease A and treatment D carries wrong relational information. That is, in other words, edge formation based on co-occurrence may lead to misleading representations.

In order to address above issue, we resort to the concept of the relative risk (RR) as a vehicle to infer the relationship and form edges accordingly ([60], [100]). Relative risk is a statistical measure of statistical method utilized in cohort studies to infer the association between an outcome and a factor.

For example, suppose 'B' is an outcome and 'A' is a factor. Then, RR(A,B) is defined as follows:

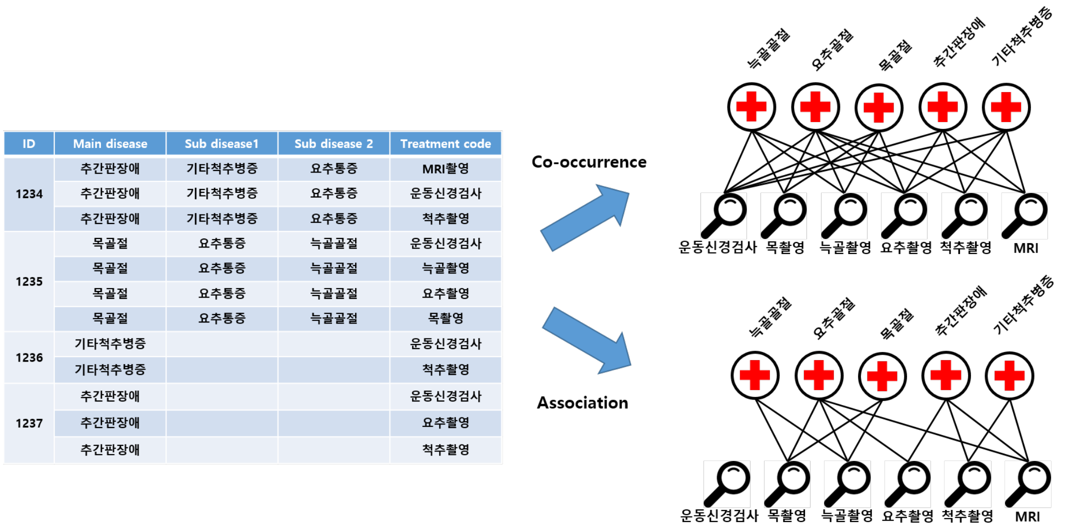
$$RR(A, B) = \frac{p(B|A)}{p(B| \sim A)}$$

If the resulting value is larger than 1, then factor 'A' is considered to be associated with outcome 'B'.

Now, we construct disease-treatment network as following. We begin by forming edges between the main diseases and the RDRG codes. If a claim has a specific RDRG code and main disease(s) listed, then edges are formed between the code and the corresponding main diseases. The main diseases, then, are connected with sub-diseases listed for the same claim, if any. Finally, we form edges between the diseases and treatments by exploiting the RR measure. More specifically, RRs are computed for all the disease-treatment pairs in the training set. Then, an edge formed for the disease-treatment pair whose RR value is greater than 1. The resulting network comprises undirected, unweighted edges.

### 4.3.2 Link Prediction between the Disease and the Treatment

In order to detect overtreatment by using node embedding vectors, we first need to select the most appropriate node embedding method which learns to represent



| ID | Main disease | Sub disease1 | Sub disease 2 | Treatment code |
|---|---|---|---|---|
| 1234 | 추간판장애 | 기타척추병증 | 요추통증 | MRI촬영 |
| | 추간판장애 | 기타척추병증 | 요추통증 | 운동신경검사 |
| | 추간판장애 | 기타척추병증 | 요추통증 | 척추촬영 |
| 1235 | 목골절 | 요추통증 | 늑골골절 | 운동신경검사 |
| | 목골절 | 요추통증 | 늑골골절 | 늑골촬영 |
| | 목골절 | 요추통증 | 늑골골절 | 요추촬영 |
| | 목골절 | 요추통증 | 늑골골절 | 목촬영 |
| 1236 | 기타척추병증 | | | 운동신경검사 |
| | 기타척추병증 | | | 척추촬영 |
| 1237 | 추간판장애 | | | 운동신경검사 |
| | 추간판장애 | | | 요추촬영 |
| | 추간판장애 | | | 척추촬영 |

Figure 4.2: Network construction by co-occurrence and association relationship

nodes effectively from the constructed network as vectors on the embedding space. In this subsection, we detail the process of choosing the best embedding method among other candidates, which constitutes two main steps: edge sampling and link prediction.

**Edge sampling**

We begin by spliting the edge set from the original network $G = (V, E)$ into two sub-graphs: the training set, $G^{trn} = (V, E^{trn})$ and the test set, $G^{tst} = (V, E^{tst})$. Since $E^{trn}$ represents all the observable, hence positive, samples, we re-denote $E^{trn}$ as $E^{trn}_{pos}$. In contrast, negative samples are not directly observed from the claims data. Thereupon, we define negative edges as the set of all the combination pairs between the diseases and treatments in the training set that are not in $E^{trn}_{pos}$. Then, sample several negative edges from this set. The number of negative edges sampled should be equal to $\left|E_{pos}^{trn}\right|$. This set of sampled negative edges are denoted by $E_{neg}^{trn}$. Similarly, we define the set of edges observed in the test set, $E^{tst}$, as the positive edge samples and denote them by $E_{pos}^{tst}$. Negative edges are sampled in a similar fashion as explained above for the training set negative samples, which is denoted by $E_{neg}^{tst}$. This validates that the set of negative edges sampled for the test set will not intersect with those in $E_{pos}^{trn}$, $E_{neg}^{trn}$, nor in $E_{pos}^{tst}$.

**Link prediction**

At this stage, we employ a selection of node embedding methods to learn to represent nodes from $G^{trn}$ as vectors on an embedding space. Suppose an arbitrary embedding model learns to present nodes $u, v$ in the $G^{trn}$, which are connected by the edge $e = (u, v)$. We denote the corresponding embedding vectors for the nodes $u, v$ by $\boldsymbol{x_u}, \boldsymbol{x_v}$, respectively. The embedding vector for the corresponding edge $e$ is defined as $\boldsymbol{x_e} = [\boldsymbol{x_u}, \boldsymbol{x_v}]$, which results from concatenating the embedding vectors of the connected nodes. We denote the sets of the embedding vectors of the edges in $E_{pos}^{trn}, E_{neg}^{trn}$ by $X_{pos}^{trn}, X_{neg}^{trn}$, respectively. Similarly, the sets of embedding vectors of edges in $E_{pos}^{tst}, E_{neg}^{tst}$ are denoted by $X_{pos}^{tst}, X_{neg}^{tst}$, respectively. Then, using $X_{pos}^{trn}, X_{neg}^{trn}$, we train a logistic regression model, $h_\theta(e)$, using to learn to classify whether a given edge is positive or negative. Finally, we evaluate the classification result of $h_\theta(e)$ with $X_{pos}^{tst}, X_{neg}^{tst}$.

The entire process for disease-treatment link prediction is illustrated in Figure 4.3. We repeat this process several times and compute the average performance of select embedding methods in solving the link prediction task. Details on the models employed in the experiment can be found in section 4.4.

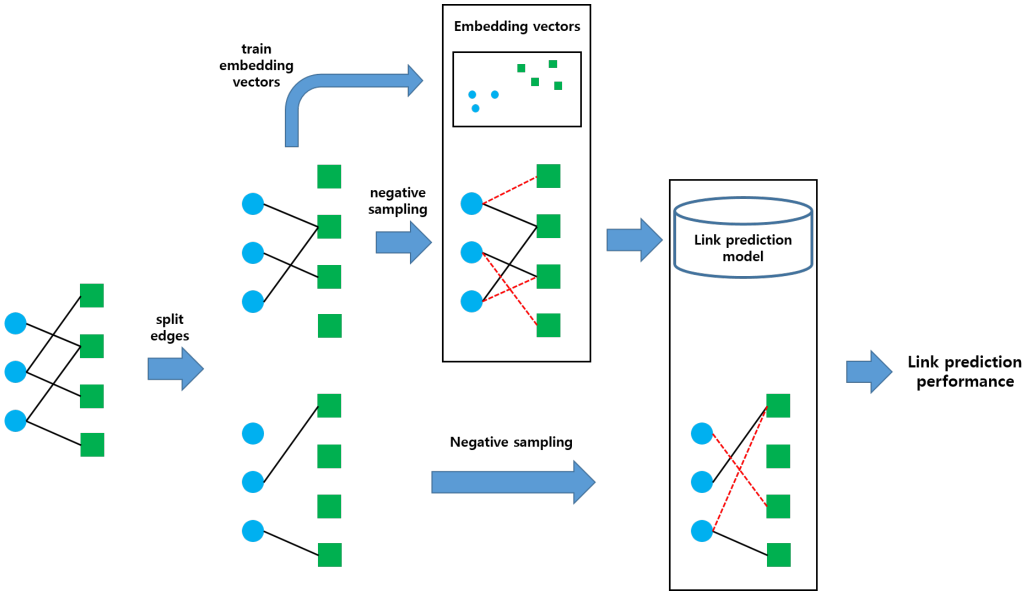


Figure 4.3: The process of link prediction between the disease and the treatment

### 4.3.3 Overtreatment Detection

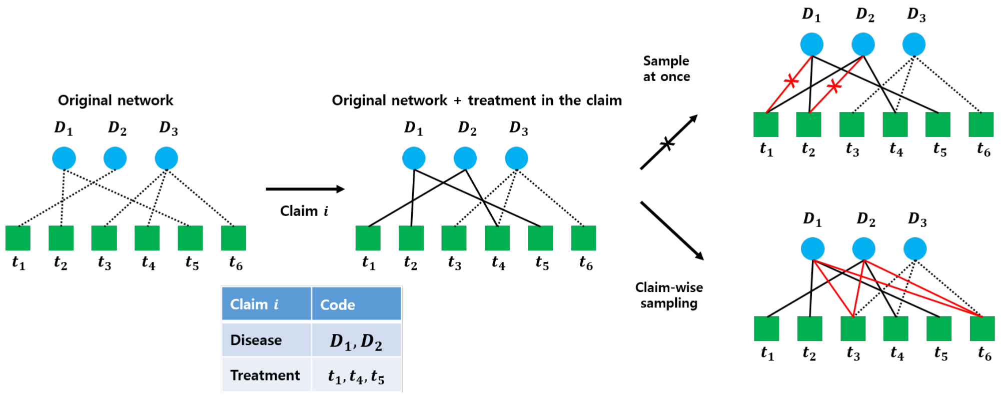
In the previous subsection, we detailed out the process for choosing the best network embedding model by comparing the average performance in disease-treatment link prediction. In this subsection, we elaborate on the framework of our overtreatment detection model which utilizes the embedding vectors of nodes. Overtreatment detection model involves two stages: edge sampling and overtreatment detection.

**Edge sampling**

Different from the previous subsection, We define the network corresponding to the training set as $G^{trn} = (V, E^{trn})$; the network corresponding to the test set, $G^{tst} = (V, E^{tst})$. Note that the nodes that do not appear during the training were also removed from the test set, hence the node set for the training is exactly what is used for the test set. The negative edge sampling process must be differentiated from what was defined in the previous subsection so as to reflect that, at this time, we need disease-treatment pair samples based on association. Take, for an example, a claim $i$ which includes diseases $d_{i1}^{trn}, d_{i2}^{trn}$ and treatment $t_{i1}^{trn}$. It may be the case that the prescription of $t_{i1}^{trn}$ is due to $d_{i1}^{trn}$, while $d_{i2}^{trn}$ is irrelevant. Such a relation should be translated to an edge $e_1 = (d_{i1}^{trn}, t_{i1}^{trn})$. On the other hand, an edge $e_2 = (d_{i2}^{trn}, t_{i1}^{trn})$ would provide misleading information, hence should not be formed. In case of subsection 4.3.2, the challenge is not as severe since $e_2$ can be sampled as a negative edge while learning to classify the disease-treatment relationship. However, in terms of evaluating the claims for overtreatment, edge sampling based on co-occurrence leads to a serious problem, since co-occurrence does not necessarily imply association. Mistakenly connecting $t_{i1}^{trn}$ to $d_{i2}^{trn}$ may lead to mis-labeling the claim $i$ as an overtreatment case, while it actually is not, for $t_{i1}^{trn}$ was an appropriate choice of prescription in response to $d_{i1}^{trn}$. Given that our ultimate goal is to detect overtreatment, such mis-labeling problem will cause grave degradation of our detection model.

In order to tackle this issue, we sample negative edges claim-by-claim, unlike the edge sampling described in the previous subsection where edges were sampled from the entire network all at once. Our negative edge sampling process preceeding the overtreatment detection proceeds as follows. Given a single claim, we identify all the diseases included in the claim. Then, we look up treatments, from the rest of the claim data set, that are not matched with either of the identified diseases. All possible combinations of the identified diseases (from the subject claim case) and the looked up treatments (from the rest of the claims data) are considered as negative edges candidates of the given claim. sample as many negative edges as the number of positive edges found in the claim.

We describe the process with an example. Suppose a claim includes diseases of $d_{i1}^{trn}, d_{i2}^{trn}$. All the treatments that are related to $d_{i1}^{trn}$ and those to $d_{i2}^{trn}$ are represented as $N(d_{i1}^{trn}), N(d_{i2}^{trn})$, respectively. Then, the negative edges are sampled from the set $E_i^{neg} = \{(u, v)|u \in \{d_{i1}^{trn}, d_{i2}^{trn}\}, v \notin \{N(d_{i1}^{trn}) \cup N(d_{i2}^{trn})\}$ where $u$ represents the disease node and $v$, the treatment node. Here, treatment nodes matched with the disease nodes from corresponding to claim $i$ is not in fact related to any of the given diseases. We denote the edge set from the training set as $E_{pos}^{trn}$; those from the test set, $E_{pos}^{tst}$. Similarly, the negative disease-treatment edge set that sampled for the disease nodes found in the training set is denoted by $E_{neg}^{trn}$; for those found in the test set, $E_{neg}^{tst}$.


Original network
$D_1$ $D_2$ $D_3$
$t_1$ $t_2$ $t_3$ $t_4$ $t_5$ $t_6$
Claim $i$
Original network + treatment in the claim
$D_1$ $D_2$ $D_3$
$t_1$ $t_2$ $t_3$ $t_4$ $t_5$ $t_6$
Sample at once
Claim-wise sampling
$D_1$ $D_2$ $D_3$
$t_1$ $t_2$ $t_3$ $t_4$ $t_5$ $t_6$
$D_1$ $D_2$ $D_3$
$t_1$ $t_2$ $t_3$ $t_4$ $t_5$ $t_6$

| Claim $i$ | Code |
|---|---|
| Disease | $D_1, D_2$ |
| Treatment | $t_1, t_4, t_5$ |

Figure 4.4: Claim-wise negative edge sampling

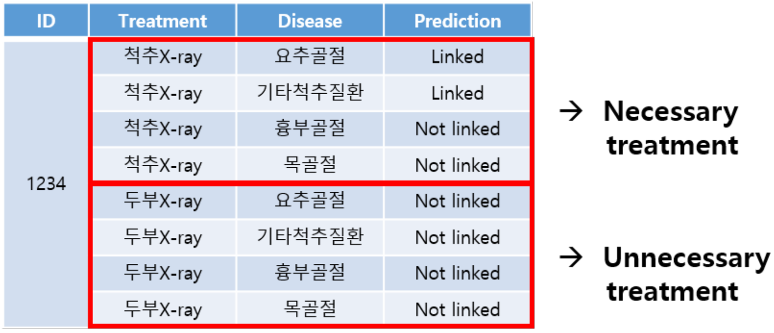
### Overtreatment detection

We propose to detect overtreatment in two different ways. First approach is to detect overtreatment naively, using the network resulting from the training set, per se. Another approach involves training node embedding models. Both approaches assume that a given case is associated with overtreatment if the link between the diseases and the treatments listed in the claim. The naïve network approach proceeds as follows: if there is a disease-treatment edge in $E_{pos}^{tst}$ which does not exists in $E_{pos}^{trn}$, the corresponding treatment is classified as overtreatment. For example, suppose that a claim includes diseases $d_1, d_2, \dots, d_m$ and a treatment $t_k$ in the test set. If the treatment satisfies the condition of $(d_1, t_k), (d_2, t_k), \dots, (d_m, t_k) \notin E_{pos}^{trn}$, we classify the subject treatment as overtreatment.

In case of node embedding approach, we first train our model to learn to vector representations of nodes that are connected by the edges in $E_{pos}^{trn}$. Then, a logistic regression model $h_\theta(e)$ is employed to learn to classify edges using $E_{pos}^{trn}$ and $E_{neg}^{trn}$. We test the training results using $E_{pos}^{tst}$ and $E_{neg}^{tst}$. If the test result reports that all of the diseases are not connected to a treatment found in the given claim, and the corresponding treatment is considered as overtreatment. Mathematically, we denote the embedding vectors of diseases $d_1, d_2, \dots, d_m$,by $\boldsymbol{x_{d_1}}, \boldsymbol{x_{d_2}}, \dots, \boldsymbol{x_{d_m}}$, and the embedding vector of the treatment $t_k$ by $\boldsymbol{x_{t_k}}$. Given the treatment $t_k$, if the prediction results for each respective diseases included in a given claim, $h_\theta([\boldsymbol{x_{d_1}}, \boldsymbol{x_{t_k}}]), h_\theta([\boldsymbol{x_{d_2}}, \boldsymbol{x_{t_k}}]), \dots, h_\theta([\boldsymbol{x_{d_m}}, \boldsymbol{x_{t_k}}])$ are all negative, then treatment $t_k$ is classified as overtreatment. We graphically illustrate the overall framework of our overtreatment detection model in Figure 4.6.

## 4.4 Experiments

In order to evaluate our proposed model, we experiment on real-world data. Our dataset consists of health insurance claims submitted to HIRA in 2017. Subsection



| ID | Treatment | Disease | Prediction |
|---|---|---|---|
| 1234 | 척추X-ray | 요추골절 | Linked |
| | 척추X-ray | 기타척추질환 | Linked |
| | 척추X-ray | 흉부골절 | Not linked |
| | 척추X-ray | 목골절 | Not linked |
| | 두부X-ray | 요추골절 | Not linked |
| | 두부X-ray | 기타척추질환 | Not linked |
| | 두부X-ray | 흉부골절 | Not linked |
| | 두부X-ray | 목골절 | Not linked |

→ Necessary treatment

→ Unnecessary treatment

Figure 4.5: Unnecessary treatment detection by link prediction result

4.4.1 provides detailed descriptions of our dataset. Subsection 4.4.2 presents the training details.

### 4.4.1 Data Description

As described in subsection 2.4.1, there are several databases separately stored within the HIRA data warehouse. Each database stores important information about insurance claims such as basic claim information, treatment information, disease information, and the filing review details. We provide details on each database in Table 2.1. From claims filed to HIRA in 2017, we extracted records that are manually reviewed. Also, we selected cases assigned to one of the following five 3-digit DRG codes for modeling and evaluation: B60(quadriplegia, paraplegia, and spondylopathy), B63(Parkinson disease, neurological neoplasm, hemiplegia, degenerative nervous system disorder), D64(disequilibrium, otitis media, upper respiratory infections), I07(simple spinal surgery, intervertebral disc removal), I68(non-surgical cervical and spinal conditions). Table 4.3 reports summary statistics of each DRG code group.

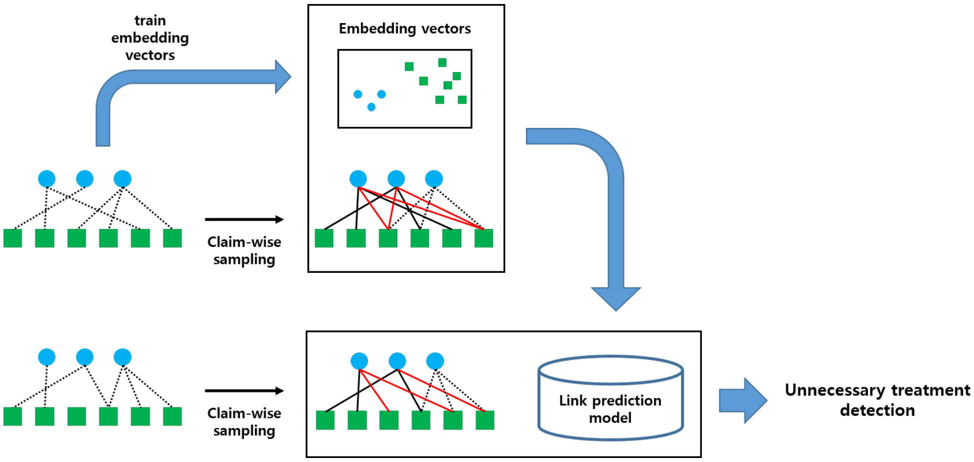


Figure 4.6: The process of unnecessary treatment detection by link prediction between the disease and the treatment

Table 4.3: Treatment data statistics

| Treatment type | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| Procedure | 804,306 | 1,562,496 | 1,951,333 | 4,504,776 | 10,586,623 |
| Prescription | 275,559 | 524,895 | 1,073,685 | 2,393,032 | 3,950,746 |
| Material | 17,384 | 16,947 | 32,712 | 509,183 | 40,753 |

1902 3-digit disease codes and 9765 treatment codes were included in the data set. Treatments are conventionally categorized into four different groups: basic treatments, procedure, prescription, and materials. Basic treatment refers to the group of treatments that may be prescribed anytime, regardless of types of diseases a patient is diagnosed with. For example, consultation, admission, nursing, and meals fall into this group. Since these types of treatments does not provide any meaningful information in relation to diseases, we discard them as we construct the disease-treatment network. Procedures are include treatments practitioners conduct on patients, such as X-ray examinations, MRI examinations, and surgical operations. Prescription refers to the detailed information about drugs practitioners prescribe such as, for example, the nonsteroidals anti-inflammatory drug. Finally, material is a group of treatments that require materials for recovery Orthosis is a good example of the material treatment. There are 5854, 2319, 1225 treatment codes in the procedure, prescription, material groups, respectively.

### 4.4.2 Experimental Settings

Our proposed method extracts disease-treatment relationship carefully by incorporating all the information available in the claim filing, instead of relying on simple co-occurrence. Multiple stages exploiting different information build up to the final disease-treatment network stage-by-stage. We begin by constructing networks separately for each DRG 3-digit code. Given a DRG 3-digit code, we extracted relevant RDRG codes, disease codes, procedure codes, prescription codes, and material codes that appear in our data. Then, we set each of these codes as individual nodes. RDRG code is of the finest granularity for DRG code in the KDRG code system. Then, for each RDRG code, we extracted the main disease codes and sub disease codes from all the claims with the matching the RDRG codes. Then, we connect the RDRG codes with the matching main disease codes. At the same time, we formed links between the main disease code nodes and the relevant sub-disease code nodes. Finally, we add treatment codes to the network and connect them with the associated disease codes as detailed in subsection 4.3.1. The diseases codes are grouped as detailed in subsection 4.4.1. Then the resulting network, by design, ensures that every treatment node is assigned to one of the three labels: procedure, prescription, or material. We present the possible types of disease-treatment pair edges in the resulting network in Table 4.4. We also provide the graphical snapshot of the network in Figure 4.7.

Table 4.4: Edges in the network and their type

| Edge | Type |
|---|---|
| RDRG - main disease | co-occurrences |
| main disease - sub diseases | co-occurrence |
| diseases - procedures | association |
| diseases - prescriptions | association |
| diseases - material | association |

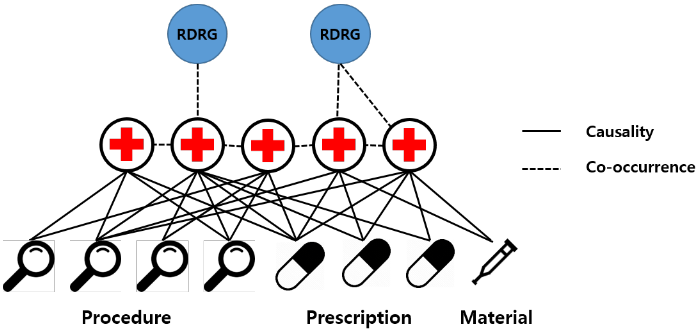


Figure 4.7: RDRG-disease-treatment network

In order to choose the best node embedding model to carry out the disease-treatment link prediction task, we trim the network by looking up the claims filed from January 2017 to September 2017 only. We split the edge sets into the training and the test set by the ratio of 7:3. The node embedding models we experimented with are as following: GF [2], HOPE [69], GraRep [9], DeepWalk [72], node2vec [29], metapath2vec [23], SDNE [92], LINE [86].

We report performances of various link prediction models employed, of which the respective unit link is defined to connect the disease and the corresponding procedure, the disease and the prescription, or the disease and the materials. We repeated the experiment for link prediction 10 times and report the average of the accuracy measures as the principal reporting metric for the overall performance. We choose the model with the highest average performance as the best embedding method for our network.

As for the overtreatment detection model as described in subsection 4.3.3, the training and the validation sets were built from the claims filed from January 2017 to September 2017. The test set comprises claims filed to HIRA from October 2017 to December 2017. We repeated the experiment 10 times and report the average accuracy as the performance metric.

We set hyper-parameters as follows. First, we fix the dimension of embedding vectors at 32 in every case. We tested various configurations and landed on the values reported. We set both the number of random walk per node and the length of each random walk at 32 for DeepWalk and node2vec methods. In case of node2vec, we set the two key hyper-parameters $p, q$, which, altogether, generate a biased random walk, at $p = 0.5, q = 2.0$. On the other hand, metapath2vec requires to define the meta-paths in order to generate random walks. We set the meta-paths to be either 'Treatment-Disease-Treatment', or 'Treatment-Disease-RDRG-Disease-Treatment'. The former meta-path implies that 'treatments caused by the same disease', while the latter, 'treatments for the same kind of patients'. We also fix the number of the random walk per node at 32. For SkipGram we set the window size for training at 6. For LINE, which considers information of neighboring nodes up to the 2nd order proximity, we set the negative ratio at 5. For GF, we fix parameter for the regularization term of the L2-norm loss at 0.00001. We used the 2-step transition probability matrix for GraRep method. The auto-encoder spart of the SDNE model takes the structure of $[n-256-32-256-n]$, where $n$ is the input dimension. Training batch size was set at 128 with the learning rate equal to 0.01. The parameters $\alpha, \beta$ in the loss function and $\nu$ in the regularization term is set to $\alpha = 0.1, \beta = 1.1, \nu = 0.3$, respectively We used pytorch [70], scikit-learn [71], scipy [91], numpy [67] packages to implement the aforementioned models.

## 4.5 Results

### 4.5.1 Network Construction

In this subsection, we compare the results between two distinct approaches to construct the disease-treatment network: (1) the simple co-occurrence-based approach, and; (2) the association-based approach. In previous subsections, we have claimed that simple co-occurrence per claim filings does not necessarily imply association. In this subsection, we will provide empirical justification for our arguments.

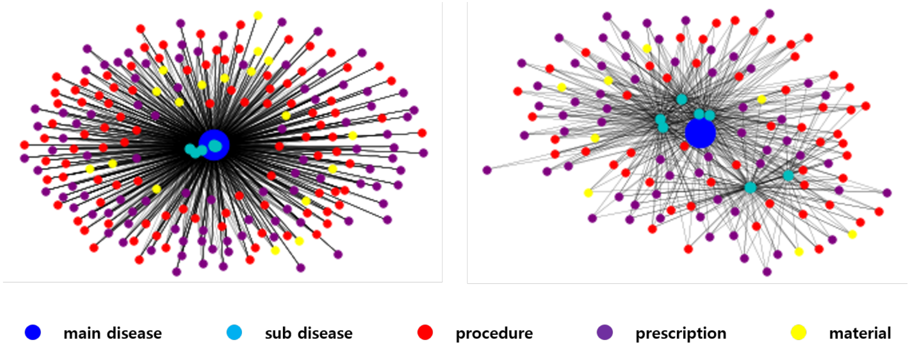


Figure 4.8: Networks constructed by co-occurrence and association relationship. (left): Co-occurrence (right): Association

Figure 4.8 represents the relationship between the main disease with code S12 (fracture of neck) and relevant sub-diseases from claim reports labeled with the DRG 3-digit code I07, whose linkage was determined by the two different approaches aforementioned. The left-hand side panel of Figure 4.8 shows the graphical representation of the relationships between the main disease S12 and the sub-diseases based on the simple co-occurrence, while the right-hand side panel shows that based on association. From now on, we refer to the left-hand side and the right-hand side network as the co-occurrence network and the association network, respectively.

In case of the co-occurrence network, every treatment which co-occurred with the main disease S12 under the DRG 3-digit code of I07 is linked not only to the main disease S12 but as well as to all the sub diseases which appeared with S12, as apparent on the left-side panel of Figure 4.8. On the contrary, treatment nodes from the association network clearly appear to be distributed more sparsely across the sub-diseases linked to the main disease S12, as shown on the right-side panel of Figure 4.8. We do not have the privilege of disclosing all the treatment nodes presented in Figure 4.8 due to personal information protection issues; yet, with permission from the appropriate authorities, we take an example from each network to provide an empirical justification for our argument. From the co-occurrence network, we have found that a link was formed between the main disease node noting a neck fracture with the treatment node for lumbar spine imaging. It is easy to see that there is no clear connection between the disease and the treatment mentioned. It is most likely due to the case in which a patient whose main disease of diagnose was the neck fracture, while lumbar spine imaging was prescribed for one of the sub-diseases not directly related or caused by the main disease.

On the contrary, for the association network, the number of direct linkages between the treatment nodes and the main disease node is far less than that of the co-occurrence network, while the different treatment types are dispersed throughout the range of the sub-diseases. Not surprisingly, the linkage between the fracture of neck and the lumbar spine imaging was missing from the association network.

### 4.5.2 Link Prediction between the Disease and the Treatment

In this subsection, we report the result of link prediction via node embedding. We trained a selection of node embedding models to learn vector representations for the nodes from the disease-treatment network and compared performance on solving the link prediction problem using the learned embedding vectors, the methodology of which is described in detail in subsection 4.3.2. Table 4.5, Table 4.6, and Table 4.7 reports each model's performance on the link prediction task using the disease-procedure, the disease-prescription, and the disease-material relations, respectively. In all cases, metapath2vec outperforms other models. This may be due to the characteristics peculiar to the network. Our network is constructed from a variety of information covering a rich range of different aspects of health insurance claims, hence strongly heterogeneous in nature. It is made of different types of nodes and edges. While other methods are devised for networks of homogeneous nature, the metapath2vec is designed to work well as heterogeneous network.

Based on the results from the link prediction test, we have selected the metapath2vec, of which the resulting embedding vectors are to be fed to the overtreatment detection model. The details on the overall mechanism of overtreatment model utilizing node embedding is elaborated in described in subsection 4.3.3.

Table 4.5: Link prediction results of disease-procedure

| Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| HOPE | 0.7337 | 0.7570 | 0.7768 | 0.7531 | 0.7662 |
| SDNE | 0.5769 | 0.5568 | 0.6162 | 0.3877 | 0.6357 |
| node2vec | 0.7426 | 0.7539 | 0.7795 | 0.7596 | 0.7720 |
| GraRep | 0.7338 | 0.7548 | 0.7798 | 0.7651 | 0.7747 |
| LINE | 0.7222 | 0.7386 | 0.7688 | 0.7609 | 0.7680 |
| GF | 0.1906 | 0.1970 | 0.1871 | 0.2037 | 0.1915 |
| DeepWalk | 0.7402 | 0.7534 | 0.7779 | 0.7621 | 0.6357 |
| metapath2vec | **0.8270** | **0.8357** | **0.8610** | **0.8534** | **0.8478** |

Table 4.6: Link prediction results of disease-prescription

| Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| HOPE | 0.7864 | 0.7927 | 0.7973 | 0.7808 | 0.8078 |
| SDNE | 0.6790 | 0.6532 | 0.6770 | 0.5320 | 0.7341 |
| node2vec | 0.7762 | 0.7890 | 0.7898 | 0.7731 | 0.8064 |
| GraRep | 0.7851 | 0.7907 | 0.8001 | 0.7847 | 0.8148 |
| LINE | 0.7794 | 0.7881 | 0.7896 | 0.7783 | 0.8063 |
| GF | 0.1701 | 0.1638 | 0.1732 | 0.1734 | 0.1672 |
| DeepWalk | 0.7721 | 0.7865 | 0.7879 | 0.7709 | 0.8067 |
| metapath2vec | **0.8155** | **0.8214** | **0.8297** | **0.8241** | **0.8487** |

### 4.5.3 Overtreatment Detection

Table 4.8 reports the performance test for the overtreatment detection. The term ‘without embedding’ refers to overtreatment detection models which utilizes the network resulting from the training set per se. The ‘proposed method’ refers to the overtreatment detection models which exploit node embedding methods to solve the link prediction problem, the mechanism of which is elaborated in detail in subsection 4.3.3. For most of the cases, the proposed model outperforms the ‘without embedding’ model. This may potentially imply that our proposed model performs better when some relational patterns are found only in the training set or in the test set, hence classifying freshly encountered treatments better.

Table 4.7: Link prediction results of disease-material

| Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|
| HOPE | 0.8328 | 0.8682 | 0.8961 | 0.7892 | 0.8679 |
| SDNE | 0.7122 | 0.7436 | 0.8563 | 0.5560 | 0.7340 |
| node2vec | 0.8511 | 0.8783 | 0.9110 | 0.7914 | 0.8881 |
| GraRep | 0.8363 | 0.8674 | 0.8991 | 0.7978 | 0.8756 |
| LINE | 0.8288 | 0.8607 | 0.8939 | 0.7878 | 0.8779 |
| GF | 0.7122 | 0.7436 | 0.8563 | 0.5560 | 0.7340 |
| DeepWalk | 0.8424 | 0.8753 | 0.9103 | 0.7992 | 0.8874 |
| metapath2vec | **0.9359** | **0.9467** | **0.9655** | **0.9038** | **0.9538** |

Table 4.8: Unecessary treatment detection by the network only and embedding vectors from the network

| Type of treatment | Method | B60 | B63 | D64 | I07 | I68 |
|---|---|---|---|---|---|---|
| Procedure | Without embedding | 0.9365 | 0.9346 | 0.9231 | 0.9387 | 0.8784 |
| | Proposed method | **0.9632** | **0.9667** | **0.9591** | **0.9768** | **0.9719** |
| Prescription | Without embedding | 0.8950 | 0.8844 | 0.9041 | 0.8983 | 0.8383 |
| | Proposed method | **0.9331** | **0.9010** | **0.9386** | **0.9238** | **0.9367** |
| Material | Without embedding | 0.8677 | **0.9230** | 0.8761 | 0.9168 | 0.8949 |
| | Proposed method | **0.9469** | 0.9177 | **0.8820** | **0.9547** | **0.8985** |

## 4.6 Summary

In the previous chapters, we proposed models for detecting abuse in medical treatments. These models, however, have yet to consider the relationship between diseases and treatments explicitly. Accounting for the disease-treatment relationship is important in a sense that, without doing so, detection models cannot properly process different drugs that have similar efficacy. There may be cases when different practitioners prescribe different drugs to a patient, where these drugs targets to alleviate the symptoms of the same disease. In order to process such cases appropriately, detection models need to be able to learn the intricate relationship between diseases and treatments.

This chapter presents a network-based approach through which the relationship between the diseases and treatments is considered during the abuse detection process. Our proposed model consists of three stages. During the first stage, a disease-treatment network is constructed based on information from claim filings. Since the association between diseases and treatments is not explicitly expressed, we infer the relationship by computing the relative risk (RR). Second stage involves selecting the best graph embedding method from several candidates available. We select the best method by comparing performances on link prediction. During the final stage, we solve a link prediction problem as the vehicle of overtreatment detection. If our link prediction model predicts links to be nonexistent for all of the diseases and treatments listed in a given claim, then the claim is classified as an overtreatment case.

We test the proposed model using the real-world claims data. Results show that the proposed method classify the treatment well which does not explicitly exist in the training network. The main contribution of this paper is that our model accounts for the disease-treatment relationship, which are not explicitly observed, during the process of overtreatment detection. Our model works well with practice patterns encountered the test phase only.

# Chapter 5

# Conclusion

## 5.1 Contribution

Abuse is a critical problem in the healthcare insurance industry. It refers to the medical service or the practice that is not consistent with the generally accepted sound fiscal practices. Reimbursing such cases cause waste of resources, eventually leading to the loss of the insurance company. Especially, abusive behaviors in national health insurance lead to social costs, which increase the premiums that the taxpayers have to pay. Therefore, detecting abuse behaviors and preventing compensation for them is a very important issue.

Currently, field professionals review the claims manually in order to screen out abuse cases. However, the astronomical increase in the number of claim filings is severely burdening the review process. Moreover, reviewing the claims require profound background knowledge and expertise, which makes the review process very costly. Adversities of such manual efforts calls for a more efficient review process. In response, past literature has employed various datamining techniques to automatically detect problematic claims or abusive providers. However, these studies do not utilize the treatment prescriptions, information of the finest granularity found in health insurance claims data. Existing studies relies on the claim-level or provider-level variables that are derived from the raw data, leading to relatively poor performance in detecting abusive claims.

The contributions of this dissertation is four-fold. Firstly, models we propose are based on medical treatment prescriptions, which is the lowest level of information available in the healthcare insurance claim. To our best knowledge, medical treatments have never been used in abuse detection. Using treatment prescriptions allows modelling abuse detection at various levels: treatment, claim, and provider-level. Secondly, we show that our finer-grained model outperforms models with higher level information. Thirdly, we propose a model which directly deals with seasonality, adding a realistic touch. Finally, we propose the abuse treatment detection model which account for the relationship between diseases and treatments, one of the most important information included in the medical treatment.

In chapter 2, we propose a scoring model based on which abusive providers are detected. Previous studies related to this topic rely primarily on provider-level variables. The coarse granularity of the mode leads to relatively poor performance. We propose the neural network-based scoring model that measures the degree of abuse for each provider. The model use treatments as input data. At the same time, we devise the evaluation metrics to quantify the efficiency of the review process. Experiment results show that the review process with the proposed model is more efficient than that with the previous model which uses the provider-level variables as input variables.

In chapter 3, we propose the method of detecting overtreatment and problematic claims under seasonality, which reflects more reality to the model. Several diseases are associated with seasonality. That is, in other words, the distribution claim is different from time to time. If the detection model does not consider this difference, its performance is not robust to the period in some departments. Instead of a single model for a department, we propose to a structure with multiple models built for several important DRG codes in the department. We test our proposed model using the real-world claim filings data, and results show that the proposed method is time-robust.

In chapter 4, we propose an overtreatment detection model accounting for the relationship between the disease and treatment. We discuss situations in which abuse detection may not work properly without the knowledge on the association relationship between disease and treatments. We propose an overtreatment detection approach method for detecting unnecessary treatment, which incorporating node embedding and link prediction methods. By solving the link prediction problem using the embedding vectors of nodes in the disease-treatment network, the model can infer pairs of disease and treatment unnecessarily reported in the insurance claims. We test our model using the real-world insurance claims data, and results show that our approach indeed works well with detecting claims with overtreatments. We additionally show that our model can be used in classifying the disease-treatment relationship.

## 5.2 Future Work

In this dissertation, we propose various abuse detection models based on the medical treatment prescription data. While our proposed models show satisfying results, there still is room for improvement. First of all, our current approach does not detect overtreatment on the claim-level. The underlying assumption here is that treatments listed in a claim are independent of one another. This may lack reality, since treatments can be prescribed to complement one another. If inference can be made on the level of individual treatments in a claim, more precise detection may be conducted.

Also, the proposed model in chapter 4 is for detecting totally unnecessary treatment, not for detecting necessary but overused treatment. In order to detect such treatment, we have to incorporate the proposed methods in this thesis. For example, train embedding vectors by graph embedding methods and train treatment classification model in chapter 2 or 3.

Finally, we can improve the performance by incorporating the data from external source. In chapter 4, we construct the disease-treatment network statistically using the claims data. However, it is unclear whether the constructed network has captured the true relationship. For example, suppose a practitioner prescribes several drugs to the patient. Some of the prescriptions may have been meant to complement each other. On the other hand, there may be the case in which the prescription includes a combination of drugs that causes side-effects when ingested together. The disease-treatment network we construct does not reflect such information. Due to the confidentiality contract, we could not utilize data from external sources as we conducted the study. However, it would help improve model performance if we could include external source data or knowledge graphs such as Drugbank[99], Twosides database [87], or SIDER [46] database.

# Bibliography

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, 2015. Software available from tensorflow.org.

[2] A. Ahmed, N. Shervashidze, S. Narayanamurthy, V. Josifovski, and A. J. Smola, *Distributed large-scale natural graph factorization*, in Proceedings of the 22nd International Conference on World Wide Web, WWW '13, New York, NY, USA, 2013, Association for Computing Machinery, p. 37–48.

[3] M. Alshahrani, M. A. Khan, O. Maddouri, A. R. Kinjo, N. Queralt-Rosinach, and R. Hoehndorf, *Neuro-symbolic representation learning on biological knowledge graphs*, Bioinformatics, 33 (2017), pp. 2723–2730.

[4] K. D. Aral, H. A. Güvenir, İ. Sabuncuoğlu, and A. R. Akar, *A prescription fraud detection model*, Computer Methods and Programs in

Biomedicine, 106 (2012), pp. 37 – 46.

[5] J. J. Baker, *Medicare payment system for hospital inpatients: diagnosis-related groups*, Journal of health care finance, 28 (2002), p. 1—13.

[6] A. Bayerstadler, L. van Dijk, and F. Winter, *Bayesian multinomial latent variable modeling for fraud and abuse detection in health insurance*, Insurance: Mathematics and Economics, 71 (2016), pp. 244 – 252.

[7] M. Belkin and P. Niyogi, *Laplacian eigenmaps and spectral techniques for embedding and clustering*, in Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, NIPS'01, Cambridge, MA, USA, 2001, MIT Press, p. 585–591.

[8] X. Cai, J. Gao, K. Y. Ngiam, B. C. Ooi, Y. Zhang, and X. Yuan, *Medical concept embedding with time-aware attention*, in Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, AAAI Press, 2018, p. 3984–3990.

[9] S. Cao, W. Lu, and Q. Xu, *Grarep: Learning graph representations with global structural information*, in Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, CIKM '15, New York, NY, USA, 2015, Association for Computing Machinery, p. 891–900.

[10] S. Cao, W. Lu, and Q. Xu, *Deep neural networks for learning graph representations*, 2016.

[11] Centers for Medicare & Medicaid Services and others, *Design and development of the diagnosis related group (DRG)*, 2018.

[12] P. S. Chan, M. R. Patel, L. W. Klein, R. J. Krone, G. J. Dehmer, K. Kennedy, B. K. Nallamothu, W. D. Weaver, F. A. Masoudi, J. S. Rumsfeld, R. G. Brindis, and J. A. Spertus, *Appropriateness of percutaneous coronary intervention*, JAMA, 306 (2011), pp. 53–61.

[13] H. Cho, B. Berger, and J. Peng, *Diffusion component analysis: Unraveling functional topology in biological networks*, in Research in Computational Molecular Biology, T. M. Przytycka, ed., Cham, 2015, Springer International Publishing, pp. 62–64.

[14] E. Choi, M. T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Tejedor-Sojo, and J. Sun, *Multi-layer representation learning for medical concepts*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1495–1504.

[15] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, *GRAM: Graph-based attention model for healthcare representation learning*, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, New York, NY, USA, 2017, Association for Computing Machinery, p. 787–795.

[16] Y. Choi, C. Y.-I. Chiu, and D. Sontag, *Learning low-dimensional representations of medical concepts*, AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science, 2016 (2016), p. 41—50.

[17] D. Clevert, T. Unterthiner, and S. Hochreiter, *Fast and accurate deep network learning by exponential linear units(ELUs)*, CoRR, abs/1511.07289 (2015).

[18] M. C. Cobanoglu, C. Liu, F. Hu, Z. N. Oltvai, and I. Bahar, *Predicting drug–target interactions using probabilistic matrix factorization*, Journal of Chemical Information and Modeling, 53 (2013), pp. 3399–3409. PMID: 24289468.

[19] G. Cybenko, *Approximation by superpositions of a sigmoidal function*, Mathematics of Control, Signals and Systems, 2 (1989), pp. 303–314.

[20] W. Dai, X. Liu, Y. Gao, L. Chen, J. Song, D. Chen, K. Gao, Y. Jiang, Y. Yang, J. Chen, et al., *Matrix factorization-based prediction of novel drug indications by integrating genomic space*, Computational and mathematical methods in medicine, 2015 (2015).

[21] J. Davis and M. Goadrich, *The relationship between precision-recall and roc curves*, in Proceedings of the 23rd International Conference on Machine Learning, ICML '06, New York, NY, USA, 2006, ACM, pp. 233–240.

[22] R. A. Derrig, *Insurance fraud*, Journal of Risk and Insurance, 69 (2002), pp. 271–287.

[23] Y. Dong, N. V. Chawla, and A. Swami, *Metapath2vec: Scalable representation learning for heterogeneous networks*, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,

KDD '17, New York, NY, USA, 2017, Association for Computing Machinery, p. 135–144.

[24] J. Duchi, E. Hazan, and Y. Singer, *Adaptive subgradient methods for online learning and stochastic optimization*, J. Mach. Learn. Res., 12 (2011), pp. 2121–2159.

[25] A. Ezzat, M. Wu, X.-L. Li, and C.-K. Kwoh, *Drug-target interaction prediction using ensemble learning and dimensionality reduction*, Methods, 129 (2017), pp. 81 – 88. Machine Learning Methods and Systems for Data-Driven Discovery in Biomedical Informatics.

[26] A. Ezzat, P. Zhao, M. Wu, X. Li, and C. Kwoh, *Drug-target interaction prediction with graph regularized matrix factorization*, IEEE/ACM Transactions on Computational Biology and Bioinformatics, 14 (2017), pp. 646–656.

[27] Federal Bureau of Investigation, *Financial crimes report to the public: Fiscal year 2010-2011*, (Date accessed: 10 October 2018), 2012.

[28] D. N. Fisman, *Seasonality of infectious diseases*, Annual review of public health, 28 (2007), p. 127—143.

[29] A. Grover and J. Leskovec, *Node2vec: Scalable feature learning for networks*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 855–864.

[30] H. He, J. Wang, W. Graco, and S. Hawkins, *Application of neural networks to detection of medical fraud*, Expert Systems with Applications, 13

(1997), pp. 329 – 336. Selected Papers from the PACES/SPICIS'97 Conference.

[31] Health Insurance Review and Assessment Service, *2017 medical expense statistics*, (Date accessed: 10 October 2018), 2018.

[32] M. T. Hecker, D. C. Aron, N. P. Patel, M. K. Lehmann, and C. J. Donskey, *Unnecessary use of antimicrobials in hospitalized patients: Current patterns of misuse with an emphasis on the antianaerobic spectrum of activity*, Archives of Internal Medicine, 163 (2003), pp. 972–978.

[33] T. Hofmann and J. Buhmann, *Multidimensional scaling and data clustering*, in Proceedings of the 7th International Conference on Neural Information Processing Systems, NIPS'94, Cambridge, MA, USA, 1994, MIT Press, p. 459–466.

[34] K. Hornik, *Approximation capabilities of multilayer feedforward networks*, Neural Networks, 4 (1991), pp. 251 – 257.

[35] Z. Huang and N. Mamoulis, *Heterogeneous information network embedding for meta path based proximity*, 2017.

[36] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, *Fasttext.zip: Compressing text classification models*, arXiv preprint arXiv:1612.03651, (2016).

[37] L. Katz, *A new status index derived from sociometric analysis*, Psychometrika, 18 (1953), pp. 39–43.

[38] S. Kim, C. Jung, J. Yon, H. Park, H. Yang, H. Kang, D. Oh, K. Kwon, and S. Kim, *A review of the complexity adjustment in the korean diagnosis-related group (KDRG)*, Health Information Management Journal, 49 (2020), pp. 62–68. PMID: 30157672.

[39] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, CoRR, abs/1412.6980 (2014).

[40] T. N. Kipf and M. Welling, *Variational graph auto-encoders*, 2016.

[41] T. N. Kipf and M. Welling, *Semi-supervised classification with graph convolutional networks*, in 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, 2017.

[42] T. Kohonen, *Self-organization and associative memory*, vol. 8, Springer Science & Business Media, 2012.

[43] Korean Financial Supervisory Service, *In 2018, 800 billion won was caught for insurance fraud and 2.4 billion won in rewards*, (Date accessed: 01 December 2019), 2019.

[44] I. Kose, M. Gokturk, and K. Kilic, *An interactive machine-learning-based electronic fraud and abuse detection system in healthcare insurance*, Applied Soft Computing, 36 (2015), pp. 283 – 299.

[45] O. Kuchaiev, M. Rašajski, D. J. Higham, and N. Pržulj, *Geometric denoising of protein-protein interaction networks*, PLoS computational biology, 5 (2009).

[46] M. Kuhn, I. Letunic, L. J. Jensen, and P. Bork, *The sider database of drugs and side effects*, Nucleic Acids Research, 44 (2015), pp. D1075–D1079.

[47] J. Lee, H. Shin, and S. Cho, *A medical treatment based scoring model to detect abusive institutions*, Journal of Biomedical Informatics, 107 (2020), p. 103423.

[48] B. E. Lehnert and R. L. Bree, *Analysis of appropriateness of outpatient CT and MRI referred from primary care clinics at an academic medical center: how critical is the need for improved decision support?*, Journal of the American College of Radiology : JACR, 7 (2010), p. 192—197.

[49] Y.-K. Lei, Z.-H. You, Z. Ji, L. Zhu, and D.-S. Huang, *Assessing and predicting protein interactions by combining manifold embedding with multiple information integration*, in BMC bioinformatics, vol. 13, Springer, 2012, p. S3.

[50] G. Li, J. Luo, Q. Xiao, C. Liang, P. Ding, and B. Cao, *Predicting microrna-disease associations using network topological similarity based on deepwalk*, IEEE Access, 5 (2017), pp. 24032–24039.

[51] X. Li, W. Chen, Y. Chen, X. Zhang, J. Gu, and M. Q. Zhang, *Network embedding-based representation learning for single cell RNA-seq data*, Nucleic Acids Research, 45 (2017), pp. e166–e166.

[52] C. Lin, C.-M. Lin, S.-T. Li, and S.-C. Kuo, *Intelligent physician segmentation and management based on kdd approach*, Expert Systems with Applications, 34 (2008), pp. 1963 – 1973.

[53] F.-M. Liou, Y.-C. Tang, and J.-Y. Chen, *Detecting hospital fraud and claim abuse through diabetic outpatient services*, Health Care Management Science, 11 (2008), pp. 353–358.

[54] Y. Luo, X. Zhao, J. Zhou, J. Yang, Y. Zhang, W. Kuang, J. Peng, L. Chen, and J. Zeng, *A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information*, Nature communications, 8 (2017), pp. 1–13.

[55] H. Lyu, T. Xu, D. Brotman, B. Mayer-Blackwell, M. Cooper, M. Daniel, E. Wick, V. Saini, S. Brownlee, and M. Makary, *Overtreatment in the united states*, PLoS One, 12 (2017).

[56] T. Ma, C. Xiao, J. Zhou, and F. Wang, *Drug similarity integration through attentive multi-view graph auto-encoders*, in Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, AAAI Press, 2018, p. 3477–3483.

[57] A. L. Maas, A. Y. Hannun, and A. Y. Ng, *Rectifier nonlinearities improve neural network acoustic models*, in Proc. icml, vol. 30, 2013, p. 3.

[58] M. E. Martinez, *The calendar of epidemics: Seasonal cycles of infectious diseases*, PLoS pathogens, 14 (2018), p. e1007327.

[59] J. M. McGinnis, L. Stuckhardt, R. Saunders, M. Smith, et al., *Best care at lower cost: the path to continuously learning health care in America*, National Academies Press, 2013.

[60] L.-A. McNutt, C. Wu, X. Xue, and J. P. Hafner, *Estimating the relative risk in cohort studies and clinical trials of common outcomes*, American Journal of Epidemiology, 157 (2003), pp. 940–943.

[61] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, 2013.

[62] T. Mikolov, W. tau Yih, and G. Zweig, *Linguistic regularities in continuous space word representations*, in HLT-NAACL, 2013, pp. 746–751.

[63] A. Mnih and G. E. Hinton, *A scalable hierarchical distributed language model*, in Advances in Neural Information Processing Systems 21, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, eds., Curran Associates, Inc., 2009, pp. 1081–1088.

[64] V. Nair and G. E. Hinton, *Rectified linear units improve restricted Boltzmann machines*, in Proceedings of the 27th International Conference on Machine Learning (ICML-10), J. Fürnkranz and T. Joachims, eds., Haifa, Israel, June 2010, Omnipress, pp. 807–814.

[65] C. Ngufor and J. Wojtusiak, *Unsupervised labeling of data for supervised learning and its application to medical claims prediction*, Computer Science, 14 (2013), p. 191.

[66] OECD, *OECD health statistics 2019*, 2019. `https://stats.oecd.org/Index.aspx?DataSetCode=SHA` (accessed on: 07 May 2020).

[67] T. E. Oliphant, *A guide to NumPy*, vol. 1, Trelgol Publishing USA, 2006.

[68] P. Ortega, C. Figueroa, and G. Ruz, *A medical claim fraud/abuse detection system based on data mining: A case study in chile*, vol. 6, 01 2006, pp. 224–231.

[69] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, *Asymmetric transitivity preserving graph embedding*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1105–1114.

[70] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, *Pytorch: An imperative style, high-performance deep learning library*, in Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.

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

[72] B. Perozzi, R. Al-Rfou, and S. Skiena, *Deepwalk: Online learning of social representations*, in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA, 2014, Association for Computing Machinery, p. 701–710.

[73] B. Perozzi, V. Kulkarni, H. Chen, and S. Skiena, *Don't walk, skip! online learning of multi-scale network embeddings*, 2016.

[74] L. Prechelt, *Early Stopping — But When?*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 53–67.

[75] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, *Struc2vec: Learning node representations from structural identity*, in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '17, New York, NY, USA, 2017, Association for Computing Machinery, p. 385–394.

[76] S. T. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science, 290 (2000), pp. 2323–2326.

[77] W. J. Rudman, J. S. Eberhardt, W. Pierce, and S. Hart-Hester, *Healthcare fraud and abuse*, Perspectives in Health Information Management/AHIMA, American Health Information Management Association, 6 (2009).

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

[79] Y. Shan, D. Jeacocke, D. W. Murray, and A. Sutinen, *Mining medical specialist billing patterns for health service management*, in Proceedings of the 7th Australasian Data Mining Conference - Volume 87, AusDM '08, AUS, 2008, Australian Computer Society, Inc., p. 105–110.

[80] Y. Shan, D. W. Murray, and A. Sutinen, *Discovering inappropriate billings with local density based outlier detection method*, in Proceedings of the Eighth Australasian Data Mining Conference - Volume 101, AusDM '09, Darlinghurst, Australia, 2009, Australian Computer Society, Inc., pp. 93–98.

[81] Z. Shen, Y.-H. Zhang, K. Han, A. K. Nandi, B. Honig, and D.-S. Huang, *mirna-disease association prediction with collaborative matrix factorization*, Complexity, 2017 (2017).

[82] H. Shin, H. Park, J. Lee, and W. C. Jhee, *A scoring model to detect abusive billing patterns in health insurance claims*, Expert Systems with Applications, 39 (2012), pp. 7441 – 7450.

[83] L. Song, C. W. Cheong, K. Yin, W. K. Cheung, B. Fung, and J. Poon, *Medical concept embedding with multiple ontological representations*, in Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press, 2019, pp. 4613–4619.

[84] M. K. Sparrow, *Fraud Control in the health care industry: Assessing the state of the art*, US Department of Justice, Office of Justice Programs, National Institute of Justice Washington, DC, 1998.

[85] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, *Dropout: A simple way to prevent neural networks from overfitting*, Journal of Machine Learning Research, 15 (2014), pp. 1929–1958.

[86] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, *Line: Large-scale information network embedding*, in Proceedings of the 24th International

Conference on World Wide Web, WWW '15, Republic and Canton of Geneva, CHE, 2015, International World Wide Web Conferences Steering Committee, p. 1067–1077.

[87] N. P. Tatonetti, P. P. Ye, R. Daneshjou, and R. B. Altman, *Data-driven prediction of drug effects and interactions*, Science Translational Medicine, 4 (2012), pp. 125ra31–125ra31.

[88] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.

[89] T. Tieleman and G. Hinton, *Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude*, COURSERA: Neural networks for machine learning, 4 (2012), pp. 26–31.

[90] G. V. Trunk, *A problem of dimensionality: A simple example*, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1 (1979), pp. 306–307.

[91] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. Jarrod Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. Vand erPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mul-

bregt, and S. . . Contributors, *Scipy 1.0: Fundamental algorithms for scientific computing in python*, Nature Methods, 17 (2020), pp. 261–272.

[92] D. Wang, P. Cui, and W. Zhu, *Structural deep network embedding*, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, New York, NY, USA, 2016, Association for Computing Machinery, p. 1225–1234.

[93] M. Wang, M. Liu, J. Liu, S. Wang, G. Long, and B. Qian, *Safe medicine recommendation via medical knowledge graph embedding*, 2017.

[94] P. Wang, T. Hao, J. Yan, and L. Jin, *Large-scale extraction of drug–disease pairs from the medical literature*, Journal of the Association for Information Science and Technology, 68 (2017), pp. 2649–2661.

[95] S. Wang, H. Cho, C. Zhai, B. Berger, and J. Peng, *Exploiting ontology graph for predicting sparsely annotated gene function*, Bioinformatics, 31 (2015), pp. i357–i364.

[96] S. Wang, E. Huang, J. Cairns, J. Peng, L. Wang, and S. Sinha, *Identification of pathways associated with chemosensitivity through network embedding*, PLOS Computational Biology, 15 (2019), pp. 1–15.

[97] S. Wang, M. Qu, and J. Peng, *PROSNET: Integrating homology with molecular networks for protein function prediction*, 2017, pp. 27–38.

[98] S.-L. Wang, H.-T. Pai, M.-F. Wu, F. Wu, and C.-L. Li, *The evaluation of trustworthiness to identify health insurance fraud in dentistry*, Artificial Intelligence in Medicine, 75 (2017), pp. 40 – 50.

[99] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda, N. Assempour, I. Iynkkaran, Y. Liu, A. Maciejewski, N. Gale, A. Wilson, L. Chin, R. Cummings, D. Le, A. Pon, C. Knox, and M. Wilson, *Drugbank 5.0: a major update to the drugbank database for 2018*, Nucleic Acids Research, 46 (2017), pp. D1074–D1082.

[100] J. Xhang and K. Yu, *What's relative risk. a method of correcting the odds ratios in cohort studies of outcomes*, Journal of the American Medical Association, 280 (1998), pp. 1690–1691.

[101] Y. Xiang, J. Xu, Y. Si, Z. Li, L. Rasmy, Y. Zhou, F. Tiryaki, F. Li, Y. Zhang, Y. Wu, X. Jiang, W. J. Zheng, D. Zhi, C. Tao, and H. Xu, *Time-sensitive clinical concept embeddings learned from large electronic health records*, BMC medical informatics and decision making, 19 (2019), p. 58.

[102] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, *Prediction of drug–target interaction networks from the integration of chemical and genomic spaces*, Bioinformatics, 24 (2008), pp. i232–i240.

[103] W.-S. Yang and S.-Y. Hwang, *A process-mining framework for the detection of healthcare fraud and abuse*, Expert Systems with Applications, 31 (2006), pp. 56 – 68.

[104] Z.-H. You, Y.-K. Lei, J. Gui, D.-S. Huang, and X. Zhou, *Using manifold embedding for assessing and predicting protein interactions from high-throughput experimental data*, Bioinformatics, 26 (2010), pp. 2744–2751.

[105] A. Zell, *Simulation neuronaler netze*, vol. 1, Addison-Wesley Bonn, 1994.

[106] X. Zeng, N. Ding, A. Rodríguez-Patón, and Q. Zou, *Probability-based collaborative filtering model for predicting gene–disease associations*, BMC medical genomics, 10 (2017), p. 76.

[107] W. Zhang, Y. Chen, D. Li, and X. Yue, *Manifold regularized matrix factorization for drug-drug interaction prediction*, Journal of Biomedical Informatics, 88 (2018), pp. 90 – 97.

[108] X. Zheng, H. Ding, H. Mamitsuka, and S. Zhu, *Collaborative matrix factorization with multiple similarities for predicting drug-target interactions*, in Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13, New York, NY, USA, 2013, Association for Computing Machinery, p. 1025–1033.

[109] M. Zitnik, M. Agrawal, and J. Leskovec, *Modeling polypharmacy side effects with graph convolutional networks*, Bioinformatics, 34 (2018), pp. i457–i466.

[110] N. Zong, H. Kim, V. Ngo, and O. Harismendy, *Deep mining heterogeneous networks of biomedical linked data to predict novel drug–target associations*, Bioinformatics, 33 (2017), pp. 2337–2344.

[111] 조성준, 신훈식, 이제혁, and 안용대, *종합병원 심사 효율화를 위한 선정모형 개선 연구*, (2018).

[112] 조성준, 이제혁, 신훈식, and 김태욱, *전문심사 선정모형 개선방안 연구*, (2018).

# 국문초록

사람들의 기대수명이 증가함에 따라 삶의 질을 향상시키기 위해 보건의료에 소비하는 금액은 증가하고 있다. 그러나, 비싼 의료 서비스 비용은 필연적으로 개인과 가정에게 큰 재정적 부담을 주게된다. 이를 방지하기 위해, 많은 국가에서는 공공 의료 보험 시스템을 도입하여 사람들이 적절한 가격에 의료서비스를 받을 수 있도록 하고 있다. 일반적으로, 환자가 먼저 서비스를 받고 나서 일부만 지불하고 나면, 보험 회사가 사후에 해당 의료 기관에 잔여 금액을 상환을 하는 제도로 운영된다. 그러나 이러한 제도를 악용하여 환자의 질병을 조작하거나 과잉진료를 하는 등의 부당청구가 발생하기도 한다. 이러한 행위들은 의료 시스템에서 발생하는 주요 재정 손실의 이유 중 하나로, 이를 방지하기 위해, 보험회사에서는 의료 전문가를 고용하여 의학적 정당성여부를 일일히 검사한다. 그러나, 이러한 검토과정은 매우 비싸고 많은 시간이 소요된다. 이러한 검토과정을 효율적으로 하기 위해, 데이터마이닝 기법을 활용하여 문제가 있는 청구서나 청구 패턴이 비정상적인 의료 서비스 공급자를 탐지하는 연구가 있어왔다. 그러나, 이러한 연구들은 데이터로부터 청구서 단위나 공급자 단위의 변수를 유도하여 모델을 학습한 사례들로, 가장 낮은 단위의 데이터인 진료 내역 데이터를 활용하지 못했다.

이 논문에서는 청구서에서 가장 낮은 단위의 데이터인 진료 내역 데이터를 활용하여 부당청구를 탐지하는 방법론을 제안한다. 첬재, 비정상적인 청구 패턴을 갖는 의료 서비스 제공자를 탐지하는 방법론을 제안하였다. 이를 실제 데이터에 적용하였을 때, 기존의 공급자 단위의 변수를 사용한 방법보다 더 효율적인 심사가 이루어 짐을 확인하였다. 이 때, 효율성을 정량화하기 위한 평가 척도도 제안하였다. 둘째로, 청구서의 계절성이 존재하는 상황에서 과잉진료를 탐지하는 방법을 제안하였다. 이 때, 진료 과목단위로 모델을 운영하는 대신 질병군(DRG) 단위로 모델을 학습하고 평가하는 방법을 제안하

였다. 그리고 실제 데이터에 적용하였을 때, 제안한 방법이 기존 방법보다 계절성에 더 강건함을 확인하였다. 셋째로, 동일 환자에 대해서 의사간의 상이한 진료 패턴을 갖는 환경에서의 과잉진료 탐지 방법을 제안하였다. 이는 환자의 질병과 진료내역간의 관계를 네트워크 기반으로 모델링하는것을 기반으로 한다. 실험 결과 제안한 방법이 학습 데이터에서 나타나지 않는 진료 패턴에 대해서도 잘 분류함을 알 수 있었다. 그리고 이러한 연구들로부터 진료 내역을 활용하였을 때, 진료내역, 청구서, 의료 서비스 제공자 등 다양한 레벨에서의 부당 청구를 탐지할 수 있음을 확인하였다.

**주요어**: 부당청구 탐지, 건강보험, 진료내역, 딥러닝, 데이터마이닝

**학번**: 2014-30327

# 감사의 글

서울대학교 산업공학과의 모든 식구들께 감사드립니다.