Negative Insurance Claim Generation Using Distance Pooling on Positive Diagnosis-Procedure Bipartite Graphs

MD ENAMUL HAQUE and MEHMET ENGIN TOZAL,
School of Computing and Informatics, University of Louisiana at Lafayette, U.S.A.

Negative samples in health and medical insurance domain refer to fraudulent or erroneous insurance claims that may include inconsistent diagnosis-procedure relations with respect to a medical coding system. Unfortunately, only a few datasets are publicly available for research in health insurance domain, yet none reports any negative claims. On the other hand, negative claims are essential not only to develop new machine learning approaches, but also to test and validate automated artificial intelligence systems deployed by insurance providers. In this study, we introduce a synthetic negative claim generation procedure based on the bipartite graph representations of positive claims. Our empirical results demonstrate promising outcomes that will improve the development and evaluation processes of machine learning approaches in healthcare, where negative samples are required, but not available. Moreover, the proposed scheme can be applied to other domains, where bipartite graph representations are meaningful and negative samples are lacking.

CCS Concepts: • **Computing methodologies → Artificial intelligence**; **Machine learning approaches**;

Additional Key Words and Phrases: Negative Health Insurance Claims, Distance Pooling, Diagnosis-Procedure Bipartite Graphs

1 INTRODUCTION

The financial facet of the healthcare industry is a complex system consisting of multiple entities working together. Figure 1 demonstrates a simplified view of the entities involved in a healthcare claim reconciliation process. Before a patient gets any service, the service provider's office confirms the patient's financial responsibilities and insurance plan. Next, the patient is checked in and the service provider examines the patient to identify the relevant diagnoses. Depending on the examination and initial diagnoses, the service provider treats the patient with one or more medical interventions, including diagnostic or surgical interventions, that are collectively called procedures. These diagnoses and procedures are typically stored in the patient's report, along with secondary and tertiary information about the patient and the visit. At this point, the patient typically checks out with or without paying a copay depending on his/her insurance plan. Then, the patient's report is sent to a medical coder who translates the information in the report into medical codes, i.e., diagnosis and procedure codes, and forms a "superbill". The medical coder electronically transfers the superbill to a medical biller. The medical biller creates a medical claim and ensures that the claim meets the coding standards and format. The claim is then sent to the patient's health insurance provider, which adjudicates the claim and decides whether the claim is valid, correct and compliant. The health insurance provider prepares a report detailing the procedures that are covered by the

Authors' address: Md Enamul Haque, enamul@louisiana.edu; Mehmet Engin Tozal, metozal@louisiana.edu,
School of Computing and Informatics, University of Louisiana at Lafayette, Lafayette, Louisiana, 70504, U.S.A..

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2022 Association for Computing Machinery.

XXXX-XXXX/2022/10-ART $15.00
https://doi.org/https://doi.org/10.1145/3531347

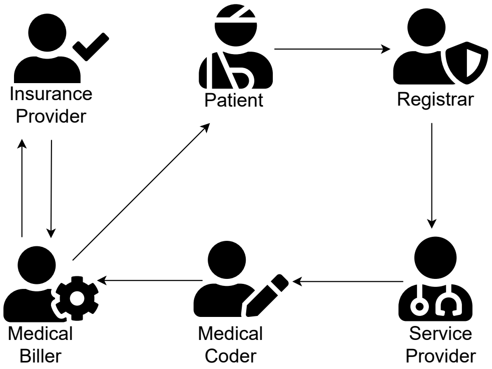


Fig. 1. A simplified view of the entities involved in healthcare claim reconciliation process [14].

patient's insurance plan and sends it back to the medical biller. Finally, the medical biller sends a statement to the patient describing the benefits, insurance coverage and remaining balances.

Unfortunately, this process is not immune to frauds. Fraudulent or fabricated claims bear a very high cost, despite they constitute a small portion of all claims in the United States. In fact, the National Health Care Anti-Fraud Association (NHCAA) reports that the financial losses due to fraudulent activities are in the orders of tens of billions of dollars in the U.S. [31]. Typical fraudulent activities include billing for more expensive procedures, fabricating claims for unperformed procedures, performing unnecessary procedures, separately billing multiple steps of a single procedure and presenting medically unnecessary procedures, e.g., cosmetic surgeries, as inevasible. These fraudulent activities often present themselves as inconsistencies between diagnoses and procedures in an insurance claim.

In AI-based healthcare insurance claim processing, the fraudulent claims are of great importance as they are necessary components to assess the accuracy of classification tasks or validate the outputs of clustering tasks. A major issue in experimenting with the fraud, on the other hand is the lack of fraudulent claim data in the publicly available datasets. Only a few datasets are publicly available for research, yet none reports *fraud* or *erroneous* cases, i.e., *negative* claims. These negative cases are essential not only to research and develop new machine learning approaches, but also to test and validate the automated Artificial Intelligence (AI) systems deployed by insurance providers.

In this study, we introduce a synthetic negative claim generation procedure based on distance pooling on bipartite graph representations of positive diagnosis-procedure codes. Moreover, we investigate the variations of the process to achieve realistic negative claims where the diagnosis and procedure codes are coherent to some desired, "quantifiable" extent. We formulate the problem over a minimal, definitive claim data consisting of procedure and diagnosis codes, because accessing richer datasets are often prohibited by law (HIPAA and GDPR) and exhibit differences among various healthcare reporting software. Moreover, medical coding systems are typically developed, governed and standardized by international organizations, while there is no standard for auxiliary information. We use Medicare and Medicaid data from Centers for Medicare and Medicaid Services (CMS) [7]. Similar to [4, 19, 20], we assume that the CMS dataset consists of valid, positive ground truth claims, which were analyzed and verified by the insurance provider before their reimbursements. The positive dataset consists of claims containing a set of diagnosis codes and a set of procedure codes, without one-to-one correspondences between them. Table 1 demonstrates a sample positive claim data containing ICD-9 (International Classification of Diseases 9th Revision) diagnosis and

Table 1. A sample positive claim data containing international ICD-9 diagnosis and procedure codes.

Diagnosis Code	ICD9v3 Description
4439	Peripheral vascular disease, unspecified
4289	Heart failure
4240	Mitral valve disorders, unspecified

Procedure Code	ICD9v3 Description
3950	Angioplasty of other non-coronary vessel(s)
4019	Other diagnostic procedures on lymphatic structures
49320	Laparoscopic Procedures on the Abdomen, Peritoneum, and Omentum
41400	Coronary atherosclerosis of unspecified type of vessel, native or graft

procedure codes. Note that the dataset contains only the codes, the code descriptions are retrieved from an online catalog [13] to augment the codes in the table.

Let $C^+ = \{c_1^+, c_2^+, \dots, c_{|C^+|}^+\}$ be the set of ground truth, positive claims and D and P be the sets of all diagnosis and procedure codes in our dataset, respectively. To generate a negative claim c_i^- from a positive claim $c_i^+ = \{D_i, P_i\}$ consisting of a set of diagnosis $D_i \subset D$ and a set of procedures $P_i \subset P$, a typical approach is to randomly decide to replace each procedure code in P_i by some probability, τ, while fixing the set of diagnosis codes, D_i. The replacement process considers all procedures in P with uniform probability distribution. Unfortunately, the uniformly at random replacement approach generates procedure codes that are highly irrelevant and inconsistent with the diagnosis codes. For example, procedure code 3950 (Angioplasty of other non-coronary vessel) in Table 1 can be replaced by procedure code 70714 (Ulcer of heel and midfoot), where the original diagnoses are related to heart and vascular problems. Our main insight is that a realistic negative claim generation process should randomly replace the procedures by the ones that are "close" or "relevant" to the original set of diagnoses to some quantifiable extent.

To quantify the closeness of diagnoses and procedures and to measure the similarity of diagnoses in terms of their treatments, we represent the relations appearing in positive claims as a bipartite graph, $G = (D, P, E)$, such that D is the set of diagnosis codes, P is the set of procedure codes and $E \subseteq D \times P$ is the set of edges representing the diagnosis and procedure codes appearing in the same claim. Figure 2 shows the partial view of the ground truth, positive bipartite graph demonstrating the claim presented in Table 1. In the figure, the vertices on the left represent the diagnosis codes, the vertices on the right represent the procedure codes and the edges represent the diagnosis-procedure code pairs appearing in the same claim. Intuitively, any two diagnoses that are similar in terms of their treatments share one or more procedures. Moreover, the hop distances between these diagnosis and their disjoint procedures are shorter in the bipartite graph compared to their distances to other procedures.

We trained a one-class SVM classifier with the positive dataset and tested our approaches using both true positive and synthetic negative claims. The empirical results show that our procedure with average distance pooling, such as average softmax, performs worse or equal for all procedure replacement probabilities compared to the baseline, uniform replacement at random, method on inpatient dataset. It is important to note that a lower accuracy in our experiments implies better performing negative insurance claims. Regarding the outpatient claims dataset, minimum distance pooling with partitional softmax and partitional proportions exhibit lower accuracies compared to the baseline approach for procedure replacement probabilities of 0.7 or lower. We observe similar results for the methods based on average distance pooling as well. In addition, we developed the truncated versions of the proposed approaches which decreases the classifier accuracy around 20%,

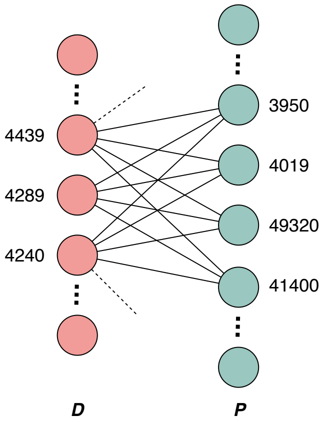


Fig. 2. A partial view of the ground truth, positive bipartite graph demonstrating the claim presented in Table 1.

on the average. Lastly, our methods incorporate diagnosis distance pooling which ensures that a procedure that is directly connected to the diagnosis group of a positive claim will not be present in the corresponding synthetic negative claim. However, baseline method with uniform procedure replacement at random does not guarantee this incident.

The synthetic negative data generation problem appears in different application domains such as DNA sequence analysis [41], image recognition [27], sentiment data generation [34], and view generation [9]. Several other data generation solutions have been introduced in different domains to alleviate the issue of imbalanced datasets [23] and missing data [21] [17] where the methods follow various sampling and data estimation techniques. To the best of our knowledge, this is the first study formulating the negative insurance claim generation problem in terms of relevant diagnosis and procedure codes and introducing a solution within the same context. The proposed scheme can also be applied to other domains to generate synthetic negative samples where positive data can be represented as bipartite graphs.

The rest of the paper is organized as follows. In Section 2 we present the related work. In Section 3 we describe the negative claim generation procedure and its variations. Section 4 demonstrates the dataset used in our experiments along with empirical evaluations. Finally, Section 5 concludes our study.

2 RELATED WORK

Negative example generation is crucial for the evaluation of classification and clustering algorithms. Typically, datasets used in learning tasks have instances of all different classes. However, there are cases where obtaining the instances of a class is not practical or simply not available. In those cases one option is to generate high quality synthetic data. In the following, we present synthetic data generation techniques in different domains such as medical image processing, computational genomics, and natural language processing areas.

Protein Sequence Analysis: Synthetic protein dataset was generated in [43] according to the structure and properties of SCOP95 (Structural Classification of Proteins) [33] protein sequence data set. The authors employed a high-speed TRIBE-MCL algorithm to validate the synthetic data. They demonstrate that the generated datasets have significant variability due to randomness in the process which is suitable to test graph-based clustering algorithms on large-scale data.

Medical Image Processing: Frid-Adar et al. developed a synthetic medical image data augmentation technique using Generative Adversarial Network (GAN) for improved liver lesion classification [15]. They specifically focused on computed tomography (CT) images of 182 liver lesions for the augmentation and used a convolutional neural network (CNN) based classifier to test the model. The experimental results demonstrate that the data augmentation improves the classification performance by 7% compared to the classical augmentation technique. Synthetic MRI images of brain tumor are generated using GAN in [40]. The authors presented two major benefits of such data generation in image segmentation and data privacy areas. Firstly, the synthetic image data helps to identify brain tumor segments with robust performance. Second, the data generation helps to share the patient tumor data with different levels of anonymization. They applied their method on two publicly available brain tumor MRI datasets to demonstrate its effectiveness. Robust and effective classification models for medical image analysis are difficult to build due to the scarcity of labeled and well-distributed image data. In [5] the authors used GAN to generate realistic and high resolution skin lesion images to help with creating labeled and non-skewed data. They used both benign and malignant dermoscopic images from ISIC2018 challenge [22] to evaluate their models.

Computational Genomics: Smith et al. [41] developed a method to generate synthetic genome data. Specifically, they improved the accurate assembly time of synthetic oligonucleotides DNA. The method was tested by establishing conditions of rapid assembly of infectious genome of bacteriophage $\phi X174$. They also proposed to use larger assembled genomes separately which will require a minimal cellular genome.

Sensory Data: Alzantot et al. [1] demonstrated a network model that uses a generator and discriminator model to distinguish between actual and synthetic accelerometer traces of smartphone. The goal of the study is to preserve the implications of data privacy for user specific data collection and analysis.

Network Analysis: Barse et al. [2] developed a method to generate synthetic log data from a set of authentic data as a seed. They specifically used Video-on-Demand (VoD) system data as the authentic set to generate different scenarios such as break-in, billing, illegal redistribution, and login fraud. The experiments demonstrated that the synthetic data can be used for training and testing a fraud detection system for the VoD services.

System Process and Policies: Cognitive autonomous assets require unforeseen context based knowledge to function properly in critical environments. Bertino et al. [6] proposed a method that helps the system assets to generate the knowledge with minimal human supervision and mediation. For instance, in a battlefield, robot armies may generate attack policy and retreat routes based on the enemy behaviors.

Natural Language Processing: Synthetic text generation is one of the active research topics in Natural Language Processing (NLP) area. Guo et al. [18] used a leaked feature extraction in the GAN based discriminator phase to guide long text generation processes. The method solves the problems of non-informativeness and sparsity compared to previous GAN based methods. Their experimental results showed that the human ratings and BLEU scores [32] are significantly better regarding longer sentences. Lin et al. [28] introduced RankGAN, a novel GAN based approach, to generate language descriptors. The discriminator is trained using a ranking function to evaluate human and machine written texts to create a better generator. They used policy gradient to optimize the RankGAN objective function. The experimental results on several publicly available dataset show the effectiveness of the method.

Medical Claim Frauds: In addition to insurance claim frauds, we also notice other fraudulent activities such as medical and prescription frauds. Ekin et. al. [12] proposed a hierarchical Bayesian method for medical fraud detection. Specifically, their method identifies the hidden relation between a provider and medical procedures. The authors also utilized sampling in overpayment estimation and medical fraud assessment in [11]. Zafari and Ekin used topic models to group drugs based on billing patterns and medical specialities [45]. The goal of this study is to assist medical auditors by providing leads for auditing providers who prescribe medically unnecessary drugs. In one of their recent studies [10], they used various classification algorithms and data proceprocessing methods on claim payment and overpayment scenarios to help healthcare professionals evaluate the merits and demerits of the analysis techniques.

Shi et.al. introduced a hybrid fraud detection approach (HFDA) which they employed in Dareway Medical Insurance Claim System, China [39]. The system mainly transformed the claims into behavior sequences and later obtained fraud probability through Dempster's rule of combination. The method relied on private claims dataset that includes claimant, hospital information, and claim approving authorities. It mainly identifies cost distribution for various diseases based on disease names for individual claimants and determines outliers. However, the method do not disclose any standard coding scheme used within the HFDA system. Therefore, adopting the HFDA system for fraud identification from claims with different coding format and standard is not possible.

Kareem et. al. demonstrated their primary stage fraudulent claim identification method based on clustering and association rule mining [25]. The authors presented different steps of their system with limited discussion on the claim data format. They also mentioned fraudulent and non-fraudulent claims wihout proper definition of each categories. They presented support and confidence scores of association rule mining applied on two example events based on disease keywords. However, the authors have not disclosed whether the keywords are extracted from clinical notes or not. Please note that the access to clinical notes are not only challenging but also the use of keywords can raise ambiguous interpretation during attribute identification and generation of association rules. Finally, they propose to use Support Vector Machine (SVM) as a supervised clustering algorithm to classify claims into fraud and non-fraud cases without desciribing the training and test data format.

Bauder et. al. [3] employed the database of Office of Inspector General List of Excluded Individuals and Entities (LEIE) which provides a major source of fraudulent providers. As claims of a provider is marked as fraudulent if that provider becomes a member of the LEIE database, the later submitted claims are automatically tagged as fraudulent. This scheme fails to identify claim level fraud analysis because not all the claims from a fraudulent provider are counterfeit. Additionally, the LEIE based fraudulent provider identification is not a real-time procedure, which hinders the instantaneous verification and validation of insurance claims without specifying the providers. Therefore, the claims based fraud identification has greater importance in terms of practice.

Several studies exist in the literature that introduces synthetic electronic health record generation technique to evaluate different classification and clustering models. Walonski et. al [44] introduced Synthea [1] at MITRE corporation to visualize and interact with synthetic patient and population health data, which is used in the state of Massachusetts, USA. However, their study does not include synthetic insurance claims, which is left for future works.

Recent studies also attempted to identify fraudulent healthcare claims from Medicare claims dataset using different techniques such as concept drift and primitive sub peer group analysis [38], sequence mining [29], weighted risk model [35], blockchain based models [36], and manifold learning [16]. However, these studies consider payment, visit sequence, frequency of disease patterns, and other transactions data to identify fraud cases. In addition, the publicly available health insurance claims data do not report *fraud* or *erroneous* cases, i.e., *negative* claims, which are essential not only to research and develop new machine learning approaches, but also to test and validate the automated Artificial Intelligence (AI) systems deployed by insurance providers.

[1] https://synthea.mitre.org/

In this work, we formulate the negative insurance claim generation problem in terms of relevant diagnosis and procedure codes and introduce a solution based on the bipartite graph representations of positive claims.

3 NEGATIVE CLAIM GENERATION PROCESS

In this section, we first formally introduce the negative data generation problem. Next, we demonstrate our overall solution approach using an algorithmic template. Finally, we detail the individual components of the algorithmic template.

3.1 Problem Statement

Let us assume that we are given a dataset of positive, ground truth insurance claims $C^+ = \{c_1^+, \ldots, c_{|C^+|}^+\}$, where $|C^+|$ is the number of claims. By "positive" claim dataset, we imply that the claims in the dataset have been analyzed, verified and reimbursed by an insurance provider. Each claim c_i consists of a set of diagnosis and procedure codes summarizing the treatment for a particular patient. Let $D = \{d_1, d_2, \ldots, d_{|D|}\}$ and $P = \{p_1, p_2, \ldots, p_{|P|}\}$ be the sets of all diagnosis and procedure codes in the dataset, where $|D|$ and $|P|$ are the number of unique diagnosis and procedure codes, respectively. The overall problem statement is that given a set of ground truth, positive claims, C^+, can we synthetically generate a set of negative claims, C^-, comprised of inconsistent or irrelevant diagnosis and procedure codes to some quantifiable extent?

In this paper, we tackle the problem from a graph theoretic perspective using statistical sampling. Our hypothesis regarding negative example generation is that the proposed method(s) should generate *realistic* negative examples which create difficulty for classifiers or domain experts to distinguish between class labels.

3.2 Problem Solution

Given a positive claim, $c_i^+ = \{D_i, P_i\}$ in C^+ such that $D \supset D_i = \{d_1, \ldots, d_k\}$, $P \supset P_i = \{p_1, \ldots, p_l\}$, consisting of k diagnoses and l procedures, a typical negative claim generation approach involves replacing the diagnosis and procedure codes randomly at uniform. The new claim, c_i^- would bring inconsistent and irrelevant procedures and diagnoses together. A closely related, alternative approach is to randomly replace only the procedure codes of c_i^+, which would also generate procedures that are inconsistent among each other, but also highly irrelevant to the corresponding diagnoses. Please note that, inconsistent procedures are fine, as long as they are relevant to their corresponding diagnoses in the diagnosis set. On the other hand, inconsistent diagnoses are natural in positive claims, as one may have multiple unrelated conditions.

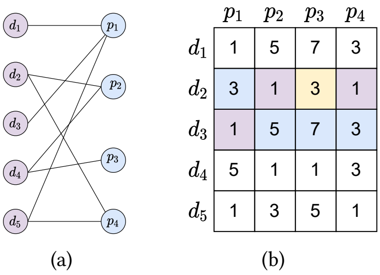
We compute a hop distance matrix $\mathcal{M}$, representing the distances between diagnosis and procedure codes. To put in other words, each row in $\mathcal{M}$ is a vector embedding of a diagnosis code in terms of its distance to all procedure codes. To generate a negative claim c_i^- from a positive claim $c_i^+ = \{D_i, P_i\}$ consisting of a set of diagnoses D_i and a set of procedures P_i, we first pool the distance embeddings of diagnosis codes, D_i, from $\mathcal{M}$ by aggregating the distances via element-wise averages or minimum. Next, we assign probabilities to each procedure based on its distance to the set of diagnoses in the claim. We developed four different probability assignment schemes, including softmax, proportions, partitional proportions and partitional softmax, where the goal is to assign relatively higher probabilities to the procedures that are closer to the set of diagnoses. Lastly, we randomly decide to replace each procedure by proabability τ. If a procedure is to be replaced, we consider all procedures in P according to the probability distribution scheme computed in the previous step.

Table 2 demonstrates an example synthetic, negative claim (right) generated from a positive claim (left) containing ICD-9 diagnosis and procedure codes. Note that the dataset contains only the codes, the code descriptions are retrieved from an online catalog [13] to augment the codes in the table. The random sampling method presented in Table 2 has a major limitation that the procedures are extremely inconsistent and/or they are ridiculously irrelevant to the corresponding diagnoses. For example, procedure code 3950 (Angioplasty of other non-coronary vessel) in Table 2 is replaced by procedure code 70714 (Ulcer of heel and midfoot), where the original diagnoses are related to heart and vascular problems. These negative examples are not only unrealistic but also easily separable from positive claims by classifiers and/or domain experts.

Table 2. An example synthetic negative claim (right) generated from a positive claim (left) containing ICD-9 diagnosis and procedure codes.

Positive Claim		**Synthetic Negative Claim**	
Diagnosis Code	**ICD9v3 Description**	**Diagnosis Code**	**ICD9v3 Description**
4439	Peripheral vascular disease, unspecified	4439	Peripheral vascular disease, unspecified
4289	Heart failure	4289	Heart failure
4240	Mitral valve disorders, unspecified	4240	Mitral valve disorders, unspecified
Procedure Code	**ICD9v3 Description**	**Procedure Code**	**ICD9v3 Description**
3950	Angioplasty of other non-coronary vessel(s)	70714	Ulcer of heel and midfoot
4019	Other diagnostic procedures on lymphatic...	4019	Other diagnostic procedures on lymphatic ...
49320	Laparoscopic Procedures on the Abdomen ...	49320	Laparoscopic Procedures on the Abdomen ...
41400	Coronary atherosclerosis of unspecified type ...	5502	Nephrostomy

Generating more realistic negative examples requires quantifying the extent of relevancy (or irrelevancy) of procedures to diagnoses. We represent the positive, ground truth dataset C^+ as a bipartite graph, $G = (D, P, E)$, such that the diagnoses and procedures are two disjoint sets D and P and an edge in E only connects a vertex in D and a vertex in P. The bipartite graph representation allows us to exploit the shortest path lengths between diagnoses and procedures as a measure of relevancy. A procedure with a shorter distance to a diagnosis, has a greater relevancy compared to a procedure with a longer distance.



	p_1	p_2	p_3	p_4
d_1	1	5	7	3
d_2	3	1	3	1
d_3	1	5	7	3
d_4	5	1	1	3
d_5	1	3	5	1

Fig. 3. An example demonstration of diagnosis and procedure code distance matrix

We illustrate the concept of distance levels using a small example in Figure 3 where we assume that the entire claim dataset contains only five diagnoses ($d_1, d_2, \ldots, d_5$) and four procedures ($p_1, p_2, \ldots, p_4$). Figure 3(a) demonstrates the example bipartite graph representation of the dataset consisting of diagnosis (left) and procedure (right) codes. The edges between diagnosis and procedure code pairs indicate that the pair appears together in a claim record in the dataset. Figure 3(b) presents the all pairs shortest paths distance matrix of the bipartite graph in Figure 3(a). Each row in the matrix denotes the vector embedding of a diagnosis in terms of its hop distance to all procedures. The distance levels in the matrix are odd integers because of the bipartiteness property of the claims. For instance, the distance from diagnosis d_1 to procedure p_3 is 7 which indicates less relevancy compared to its distance to procedure p_4. We employ the distance matrix to assign meaningful procedure replacement probabilities to different distance levels while generating synthetic negative claims from positive claims.

The synthetic negative claim generation algorithm traverses the set of ground truth positive claims. For each procedure in a positive claim, the algorithm makes a random decision to replace the procedure by a fixed probability. If the decision is to replace, the algorithm randomly replaces the procedure at uniform with another one from the set of all procedures, P. Random replacement at uniform, i.e., selecting a new procedure with $1/|P|$ probability, generates unrealistic negative claims that are easy to identify, hence we propose alternative procedure selection probability distributions based on the shortest path lengths in the bipartite graph G. The probability assignment process is called "procedure probability setting". In this study, we introduce four different procedure probability setting approaches: softmax, proportions, partitional proportions and partitional softmax. Given that a positive claim consists of a set of diagnoses along with a set of procedures without specifying the exact mappings between individual diagnoses and procedures, we consolidate the distances of the procedures to the diagnosis group. This diagnosis group distance consolidation or bundling process is called "diagnosis distance pooling". In this study we introduce two types of diagnosis distance pooling approaches: minimum and average pooling.

Algorithm 1 presents a procedural template to generate negative claims using a ground truth, positive claim dataset. The algorithm, expects a positive claim dataset, C^+, a procedure replacement probability, τ, a preferred diagnosis distance pooling approach, Ψ and a preferred procedure probability setting approach, Φ, as input and returns a negative claim dataset, C^-.

Lines 1-4 of the algorithm initializes a bi-adjacency matrix, $\mathcal{B}$, representing the positive claims as a bipartite graph of diagnoses and procedures. Similar to [42], we transform the bi-adjacency matrix $\mathcal{B}$ into an adjacency matrix $\mathcal{A}$ using Equation 1 at line 3 of Algorithm 1.

$$\mathcal{A} = \begin{pmatrix} \mathbf{0} & \mathcal{B} \\ \mathcal{B}^\top & \mathbf{0} \end{pmatrix} \tag{1}$$

Line 4 computes the all pairs shortest path matrix using Johnson's [24] algorithm. Johnson's [24] algorithm produces the distance block matrix $\mathcal{D}$ presented in Equation 2.

$$\mathcal{D} = \left[\begin{array}{c|c} D \rightsquigarrow D_{|D|\times|D|} & D \rightsquigarrow P_{|D|\times|P|} \\ \hline P \rightsquigarrow D_{|P|\times|D|} & P \rightsquigarrow P_{|P|\times|P|} \end{array} \right] \tag{2}$$

where the top right block $D \rightsquigarrow P_{|D|\times|P|}$ of the distance matrix $\mathcal{D}$ refers to the shortest distances between $|D|$ diagnoses and $|P|$ procedure pairs. Without loss of generality we only consider the top right block of the distance matrix $\mathcal{M}$ at line 5.

The algorithm traverses the set of positive claims between Lines 6 and 21. At line 7, it clones a positive claim, c_i^+, as a candidate negative claim, c_i^-. At line 9 it pools the distances of the diagnoses in c_i^- to the procedures according to the preferred pooling approach, Ψ. The result of this step is a vector of diagnosis group distances, $\mathbf{v}$ to all procedures. At line 11 the algorithm translates the distances into probabilities using the preferred procedure probability setting approach, Φ. The result of this step is a probability distribution, $\mathbf{u}$, over all procedures.

At lines 12-17, Algorithm 1 traverses the procedures of c_i^- for random replacement. At line 13 it generates a random value $p \in [0, 1]$. If p is smaller than the procedure replacement probability τ, it randomly replaces the procedure according to probability distribution $\mathbf{u}$.

Algorithm 1: An algorithmic framework for synthetic negative claim generation.

```
Input: Positive claims dataset, $C^+ = \{D, P\}$
Input: Procedure replacement probability, $\tau$
Input: Diagnosis distance pooling approach, $\Psi$
Input: Procedure probability setting approach, $\Phi$
Output: Synthetic negative claims dataset, $C^- = \{D' \subseteq D, P' \subseteq P\}$
 1: Initialization:
 2:   Build a bi-adjacency matrix $\mathcal{B}_{mxn}$ using the input dataset $C^+$
 3:   Build the adjacency matrix $\mathcal{A}$ using the bi-adjacency matrix $\mathcal{B}$
 4:   Compute all pairs shortest path matrix $\mathcal{D}$ on $\mathcal{A}$ using the Johnson's algorithm
 5:   Set $\mathcal{M}$ to the top right block-distance matrix of $\mathcal{D}$
 6: for $\forall c_i^+ \in C^+$ do
 7:   $c_i^- \longleftarrow c_i^+$
 8:   Diagnosis Distance Pooling:
 9:     $\mathbf{v} \longleftarrow$ Pool the distances to the diagnoses $D_i$ of $c_i^-$ using $\mathcal{M}$ according to $\Psi$
10:   Procedure Probability Setting:
11:     $\mathbf{u} \longleftarrow$ Assign the probabilities of all procedures $P$ using $\mathbf{v}$ according to $\Phi$
12:   for $\forall p_{ij} \in P_i$ of $c_i^-$ do
13:     $p \longleftarrow$ generate a random value between 0 and 1
14:     if $p \leq \tau$ then
15:       replace $p_{ij}$ by a randomly selected procedure from $P$
           according to probability distribution $\mathbf{u}$
16:     end if
17:   end for
18:   if at least one procedure is randomly replaced in $c_i^-$ then
19:     $C^- \longleftarrow C^- \cup c_i^-$
20:   end if
21: end for
22: return $C^-$
```

If at least one procedure is replaced in the candidate negative claim c_i^-, then it is placed in the set of negative claims at line 19. Lastly, line 22 returns the negative claim dataset C^-.

Given that the core of Algorithm 1 lies in "diagnosis distance pooling" and "procedure probability setting" employed at lines 8 and 10, we present their details in the following subsections.

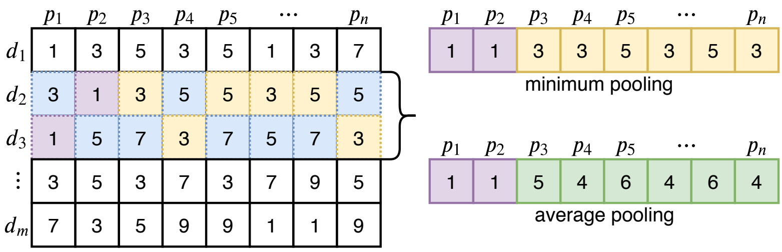
3.2.1 *Diagnosis Distance Pooling.*

Algorithm 1 clones a positive claim to create a candidate negative claim. The candidate, c_i^-, consists of a set of diagnosis, $D_i \subset D$, and set of procedures, $P_i \subset P$. The first step to generate a negative claim is to compute the distances between the diagnosis set, D_i and all procedures P, which will be translated into probabilities later. In fact, each row, $\mathbf{v}_j$, of $\mathcal{M}$ in Algorithm 1 is a distance vector corresponding to the distances between diagnosis d_j and all procedures. Given that we have multiple diagnoses in D_i we have multiple distance vectors to P. Therefore, it is necessary to pool all those distances into a single distance vector. In other words, it is necessary to compute the distances between the procedures and the group of the diagnoses rather than between the procedures and the individual diagnosis in the group. We define pooling as a element-wise aggregation function over a list of distance vectors. In the following we introduce two aggregation methods: *minimum pooling* and *average pooling*.

Minimum Pooling: Given a candidate negative claim $c_i^- = \{D_i, P_i\}$, let $D_i = \{d_1, d_2, \ldots, d_k\}$ have the corresponding distance vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$. Minimum pooling aggregates all distance vectors into a single vector $\mathbf{v}$ by taking the element-wise (or positional) minimum of the distance vectors as shown in Equation 3.

$$\mathbf{v}_i = \min\{\mathbf{v}_{1i}, \mathbf{v}_{2i}, \ldots, \mathbf{v}_{ki}\} \tag{3}$$

where $\mathbf{v}_{ji}$ is the distance of the j^{th} diagnosis to the i^{th} procedure and $\mathbf{v}_i$ is the distance of the diagnosis group to the i^{th} procedure. The interpretation of minimum pooling is that the distance of a group of diagnoses to a particular procedure should not be more than the closest diagnosis to the procedure. Figure 4 (top) demonstrates the minimum pooling concept using a diagnoses group consisting of two diagnosis.



	p_1	p_2	p_3	p_4	p_5	$\cdots$		p_n
d_1	1	3	5	3	5	1	3	7
d_2	3	1	3	5	5	3	5	5
d_3	1	5	7	3	7	5	7	3
$\vdots$	3	5	3	7	3	7	9	5
d_m	7	3	5	9	9	1	1	9

p_1	p_2	p_3	p_4	p_5	$\cdots$		p_n
1	1	3	3	5	3	5	3

minimum pooling

p_1	p_2	p_3	p_4	p_5	$\cdots$		p_n
1	1	5	4	6	4	6	4

average pooling

Fig. 4. A simple example of how minimum (top) and average (bottom) pooling are computed using Equation 3 and Equation 5.

Average Pooling: Given a candidate negative claim $c_i^- = \{D_i, P_i\}$, let $D_i = \{d_1, d_2, \ldots, d_k\}$ have the corresponding distance vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$. Average pooling aggregates all distance vectors into a single vector $\mathbf{v}$ by taking the element-wise (or positional) average of the distance vectors as shown in Equation 4.

$$\mathbf{v}_i = \frac{1}{k} \sum_{j=1}^{k} \mathbf{v}_{ji} \tag{4}$$

where $\mathbf{v}_{ji}$ is the distance of the j^{th} diagnosis to the i^{th} procedure and $\mathbf{v}_i$ is the distance of the diagnosis group to the i^{th} procedure. The interpretation of average pooling is that the distance of a group of diagnoses to a particular procedure should be the average of the distances of the individual diagnosis to the procedure. Although the average pooling approach considers all procedure distances, it masks the existence of procedures which have only one-hop distance to one or more of the diagnoses in the group. Detecting these procedures is important, because they should not be part of a negative claim as they are directly related to at least one diagnosis in the claim. Therefore we modify Equation 4 as follows:

$$\mathbf{v}_i = \begin{cases} 1 & \exists \mathbf{v}_j \text{ s.t. } \mathbf{v}_{ji} = 1 \\ \frac{1}{k} \sum\limits_{j=1}^{k} \mathbf{v}_{ji} & \text{otherwise} \end{cases} \tag{5}$$

where the distance is forced to one, if there is a diagnosis with one hop distance to a particular procedure in the group. Figure 4 (bottom) demonstrates the average pooling concept using a diagnoses group consisting of two diagnosis.

3.2.2 Procedure Probability Setting. To probabilistically replace the procedures of a candidate negative claim, c_i^-, Algorithm 1 translates its pooled distance vector, $\mathbf{v}$, into a probability distribution vector, $\mathbf{u}$. Remember that random procedure replacement at uniform, i.e., selecting a new procedure with $1/|P|$ probability, generates unrealistic negative claims with trivial decision boundaries. Therefore, we propose alternative procedure selection probability distributions, $\mathbf{u}$, based on the pooled diagnosis distance vector, $\mathbf{v}$, representing the distances between the group of diagnosis in the candidate claim and all procedures.

First of all, the distances between diagnoses and procedures only assume odd numbers, $\{1, 3, 5, \ldots\}$, because edges between any two diagnoses or procedures are not allowed in $G = (D, P, E)$, by definition. Secondly, to achieve realistic or non-trivial negative claims, the procedure probabilities should be inversely proportional to their distance to the diagnosis group. That is the farther a procedure is to the diagnosis group, the lower its selection probability. Thirdly, the procedures that have one hop distance to the diagnosis group, should assume zero probability, as they are directly related to one or more diagnosis in the group. Lastly, the probability distribution vector $\mathbf{u}$ should satisfy the Kolmogorov Axioms of Probability [26]. The first statement is only a fact. To address the second statement, we utilize the reciprocals of distances instead of the distances themselves. To put in other words, we utilize $1/\mathbf{v}_i$, rather than $\mathbf{v}_i$. To address the third statement, we define our probability assignment procedures as piecewise functions. The last statement is satisfied by the distances being positive and the necessary normalizations being applied. In the following, we introduce four different procedure probability assignment approaches satisfying our requirements: *softmax*, *proportions*, *partitional proportions* and *partitional softmax*.

Softmax: Softmax is a normalized exponential function which is frequently used in machine learning and statistics to assign probabilities to a vector of real values. Given a pooled diagnosis-procedure distance vector $\mathbf{v}$, we assign probabilities to procedures using the exponents of the reciprocals of distances. Equation 6 translates the distance vector $\mathbf{v}$ into probability vector $\mathbf{u}$ over all procedures:

$$\mathbf{u}_i = \begin{cases} 0 & \text{if } \mathbf{v}_i = 1 \\ \dfrac{\exp\left(\frac{1}{\mathbf{v}_i}\right)}{\sum\limits_{j=1}^{|P|} \exp\left(\frac{1}{\mathbf{v}_j}\right)} & \text{otherwise} \end{cases} \tag{6}$$

where $|P|$ denotes the number of procedures in the original bipartite graph block-distance matrix $\mathcal{M}$ in Algorithm 1. One particular limitation of softmax expressed in Equation 6 is that it is not scale invariant. Hence, when the real values are between 0 and 1, it may not emphasize the procedures with shorter distances as much as one may desire. In fact, this case is illustrated in Figure 5 where the probabilities that are assigned to procedures with shorter distances are roughly similar to the ones with longer distances.

Proportions: Proportional frequency or magnitude is a normalized function, which forms the basis of frequentist interpretation of probability. Given a pooled diagnosis-procedure distance vector $\mathbf{v}$, it assigns probabilities to procedures by directly using the reciprocals of distances. Equation 7 translates the distance vector $\mathbf{v}$ into probability vector $\mathbf{u}$ over all procedures:

$$\mathbf{u}_i = \begin{cases} 0 & \text{if } \mathbf{v}_i = 1 \\ \dfrac{\left(\frac{1}{\mathbf{v}_i}\right)}{\sum\limits_{j=1}^{|P|} \left(\frac{1}{\mathbf{v}_j}\right)} & \text{otherwise} \end{cases} \tag{7}$$

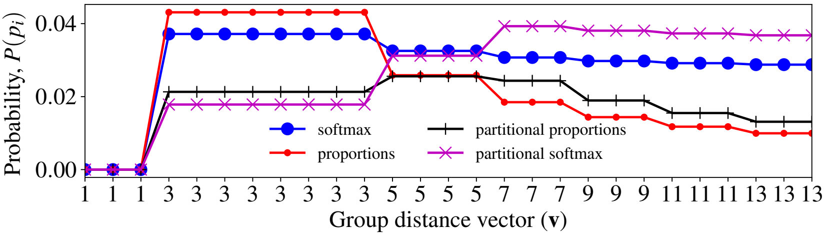


Fig. 5. Demonstration of probability distribution of procedures using group distance vector with respect to different transformations.

where $|P|$ denotes the number of procedures in the original bipartite graph block-distance matrix $\mathcal{M}$ in Algorithm 1. Figure 5 illustrates that the probabilities that are assigned to procedures with shorter distances are much higher compared to the ones with longer distances.

On the other hand, if there are many procedures with shorter distances and a few with medium or longer distances, then the ones with shorter distances overwhelmingly assume a great portion of the total probability and let the few medium or longer distance procedures starve. Figure 5 illustrates this phenomena where the procedures with distance three assume a great portion of total probability compared to the ones with distance five with proportions approach. The reason behind this phenomena is that we assign the probabilities directly to individual distances, rather than unique distance values.

Partitional Proportions: To alleviate the problem of many procedures with shorter distances assuming a great portion of the total probability, we first partition the distance vector $\mathbf{v}$ based on the unique distance levels. Let $\{l_1, l_2, \ldots, l_r\}$ be the unique distance levels (or values) of $\mathbf{v}$. Next, we assign probabilities to distance levels using the reciprocals of distance levels as presented in Equation 8

$$\mathbf{l}_i = \begin{cases} 0 & \text{if } l_i = 1 \\ \dfrac{\left(\frac{1}{l_i}\right)}{\sum\limits_{j=1}^{r}\left(\frac{1}{l_j}\right)} & \text{otherwise} \end{cases} \tag{8}$$

where l_j is the the j^{th} unique distance level and $\mathbf{l}_i$ is the likelihood assigned to the distance level l_i. Then, we equally distribute the likelihood of a distance level, among its procedures to build the probability distribution vector $\mathbf{u}$ as presented in Equation 9

$$\mathbf{u}_i = \frac{1}{|l_i|} \sum \mathbf{l}_i \mathbf{1}(l_i = \mathbf{v}_i) \tag{9}$$

where $|l_i|$ is the size of the i^{th} distance partition, $\mathbf{l}_i$ is the partitions likelihood and $\mathbf{1}(l_i = \mathbf{v}_i)$ is an indicator function that is 1 when the i^{th} procedure's distance is equal to the i^{th} partition's level. Figure 5 illustrates that procedures with distance three do not overwhelmingly assume the total probability. Note that, in the figure the probability assigned to the procedures with distance three is lower than the procedures with distance five. Although this looks like a contradiction leading to the procedures with distance five having higher selection probabilities, the higher number of procedures with distance three compensates for the case.

Partitional Softmax: Given a pooled diagnosis-procedure distance vector $\mathbf{v}$, we can also extend the partitional proportions approach to softmax. Again, we first partition the distance vector $\mathbf{v}$

based on the unique distance levels. Let $\{l_1, l_2, \ldots, l_r\}$ be the unique distance levels (or values) of $\mathbf{v}$. Next, we assign probabilities to distance levels using the exponents of the reciprocals of distance levels as presented in Equation 10

$$\mathbf{l}_i = \begin{cases} 0 & \text{if } l_i = 1 \\ \frac{\exp\left(\frac{1}{l_i}\right)}{\sum\limits_{j=1}^{r} \exp\left(\frac{1}{l_j}\right)} & \text{otherwise} \end{cases} \tag{10}$$

where l_j is the the j^{th} unique distance level and $\mathbf{l}_i$ is the likelihood assigned to the distance level l_i. Then, we equally distribute the likelihood of a distance level, among the procedures having the same distance to build the probability distribution vector $\mathbf{u}$ as presented in Equation 9 where $|l_i|$ is the size of the i^{th} distance partition, $\mathbf{l}_i$ is the partitions likelihood and $\mathbf{1}(l_i = \mathbf{v}_i)$ is an indicator function that is 1 when the i^{th} procedure's distance is equal to the i^{th} partition's level. As illustrated in Figure 5 this approach typically assigns higher probabilities to smaller number of procedures with medium or larger distances. Although we do not prefer this approach, we introduce it for the sake of completeness.

The synthetic negative insurance claim generation process presented in Algorithm 1, goes through all procedures of the candidate claim $c_i^- = (D_i, P_i)$ and replaces a procedure with probability τ and selects a replacement procedure according to the probability setting approach Φ. Obviously, the number of procedures to be replaced is random and depends on the replacement probability τ. In fact, the number of procedures to be replaced is binomially distributed. However, we consider a claim negative only if at least one procedure is replaced. Therefore, the number of procedures, X, replaced in a synthetic negative claim $c_i^- = (D_i, P_i)$ is distributed by zero-truncated binomial distribution with parameters $|P_i|$ and τ:

$$P(x) = \binom{|P_i|}{x} \frac{\tau^x (1-\tau)^{|P_i|-x}}{1-(1-\tau)^{|P_i|}} \tag{11}$$

As a result the expected value of the replaced procedures in c_i^- is:

$$E[X] = \frac{|P_i|\tau}{1-(1-\tau)^{|P_i|}} \tag{12}$$

4 EMPIRICAL EVALUATIONS

In the following, we first introduce experimental setup with a brief discussion on the process of transforming claims into vector of real numbers. Next, we discuss the datasets used in the experiments. Finally, we present experimental results on two types of health insurance claims data, namely inpatient and outpatient datasets.

4.1 Experimental Setup

Our experiemental setup consists of two steps. In the first step, we transform claims into vectors of real numbers to classify into positive and negative classes. We employ a language model on alphanumeric claim codes to embed the inherent relationships between frequently occurring diagnosis and procedure codes within the claims dataset. We specifically use Skip-gram [30] language model to transform claims into vectors. Skip-gram model uses an unordered sequence of diagnosis and procedure codes from claims dataset to compute log-likelihood of the codes within a predefined context window. The context window in our problem setup refers to the number of codes that are frequently used with individual code within the whole claims dataset. Note that the context window and vector size of the Skip-gram model are two different parameters where vector size is used to represent a code to predict its nearby codes by training a shallow neural network. Formally, given a set of diagnosis and procedure codes $\{c_1, c_2, \ldots, c_q\}$, Skip-gram model computes log-likelihood of every code c_i within a predefined context window w, as shown in Equation 13. We set the context window w as 5 to allow the Skip-gram model capture claim level representation of procedure codes. To explain the process further, we look at the Equation 13 and Figure 6 where most frequent diagnosis and procedure counts are 10. Therefore we use context window 5 to allow the model look for 5 codes on both left and right sides totaling 10 codes within its window.

$$\sum_{i-w \leq j \leq i+w} \log p(c_j|c_i) \tag{13}$$

where $p(c_j|c_i)$ denotes the conditional probability of appearing code c_j within the context window of code c_i. The conditional probability is defined using a softmax function as follows.

$$p(c_j|c_i) = \frac{\exp(V_{c_i}.U_{c_j})}{\sum\limits_{c_k \in C} \exp(V_{c_i}.U_{c_k})} \tag{14}$$

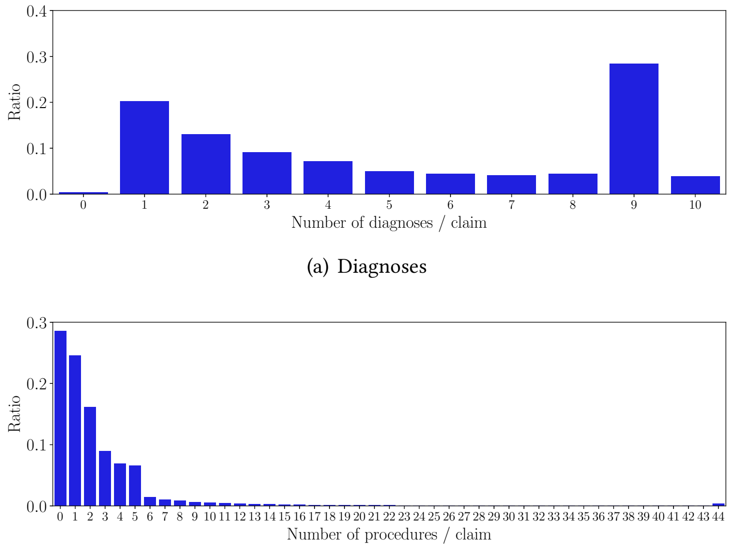
where V_{c_i} denotes T-dimensional input vector representation of code c_i, $V \in \mathbb{R}^{|C| \times T}$. Similarly, U_{c_j} denotes T-dimensional output vector representation of code c_j, $U \in \mathbb{R}^{|C| \times T}$. We defined the vector length T as 10, 20, and 50 in our experiments. We present the evaluations using vector length 20 as other lenghts exhibit similar outcomes. Once Skip-gram model is trained, we receive input code and output context code matrices. The input matrix is used to represent the codes in a defined T-dimension vector space for various predictive jobs.

A stochastic gradient descent algorithm is used to maximize the objective function in Equation 13. The computation of the objective function is expensive as the denominator in Equation 14 sums over all codes in the code vocabulary C. Mikolov et. al. [30] used negative sampling with Skip-gram to avoid the expensive computation. Once we receive the input vector representation of codes, we use that as a look-up matrix for the claim code. We used the look-up matrix to find feature representation for both training and test claims.

We employed one-class SVM [37] to classify positive and negative test claims. Usually, an SVM model is created based on splitting the training data points using a boundary with maximum gap and penalizing data points that are misclassified. The one-class SVM is trained on datasets that has only one labeled-class data. We used linear kernel with $\gamma = 0.001$ and $\nu = 0.95$ in one-class SVM during training phase on 80% positive claims. We used the rest of the positive claims and synthetic negative claims for testing. On the other hand, we used both positive and negative claims in binary class SVM model with 10-fold cross validation. The negative claims are populated for every positive claims in the dataset by following different diagnosis distance pooling and procedure probability settings (Table 3).

4.2 Datasets

We collect Medicare and Medicaid data from Centers for Medicare and Medicaid Services (CMS) website. The dataset contains inpatient and outpatient claims from years between 2008-2010, containing 20 files [7]. Each claim includes medical diagnosis and procedure codes along with de-identified patient and payment specific information. We conducted two sets of experiments to demonstrate the effectiveness of our method using 33,387 *inpatient* and 39,540 *outpatient* positive claims from two randomly selected data files. A patient is categorized as inpatient if the hospital stay is longer and prescribed by an authorized doctor for relevant procedures. Note that inpatient CMS dataset contains ICD-9 format diagnosis and procedure codes. A patient is categorized as an outpatient if he/she gets lab test, X-rays, or any other hospital services without the written order from a doctor to be admitted to a hospital as an inpatient. Outpatient dataset contains both ICD-9, HCPCS level-I (CPT), and level-II codes for the procedures.



(a) Diagnoses

(b) Procedures

Fig. 6. Combined Medicaid and Medicare claim distributions in terms of diagnosis and procedure numbers per claim.

Figure 6 summarizes the inpatient and outpatient claims in terms of the number of diagnoses and procedures present in a claim. The figure shows that a claim consists of maximum 10 and 44 diagnoses and procedures, respectively in the combined claims data. We observe that the majority of the claims share high numbers of diagnoses in the actual claims data. In addition, we observe claims that have no procedure codes. Note that our synthetically generated negative claim diagnosis and procedure frequency distribution follows similar distributions as positive claims in Figure 6, except for the claims without procedures.

4.3 Experimental Results

In this part we evaluate our negative claim generation procedures using one-class SVM. The ground truth, positive claim data is used to train a one-class SVM classification model. We use both positive and synthetically-generated negative claims for testing the model. We include additional experimental results in the supplemental file to demonstrate the need for truncated and non-truncated procedure probability settings. In addition, we present binary SVM based classification results on both inpatient and outpatient datasets.

One-class SVM is an extension of SVM which is applicable for the datasets with no negative claims. Unlike binary SVM, one-class SVM identifies negative samples by learning a decision boundary that maximizes the separation between the samples of the positive class and the rest [8, 37]. The goal is to evaluate different negative claim generation techniques using accuracy, precision, and recall on inpatient and outpatient claims data. Our evaluation also includes a uniform procedure

Table 3. Abbreviations of legends using diagnosis distance pooling and procedure probability setting functions.

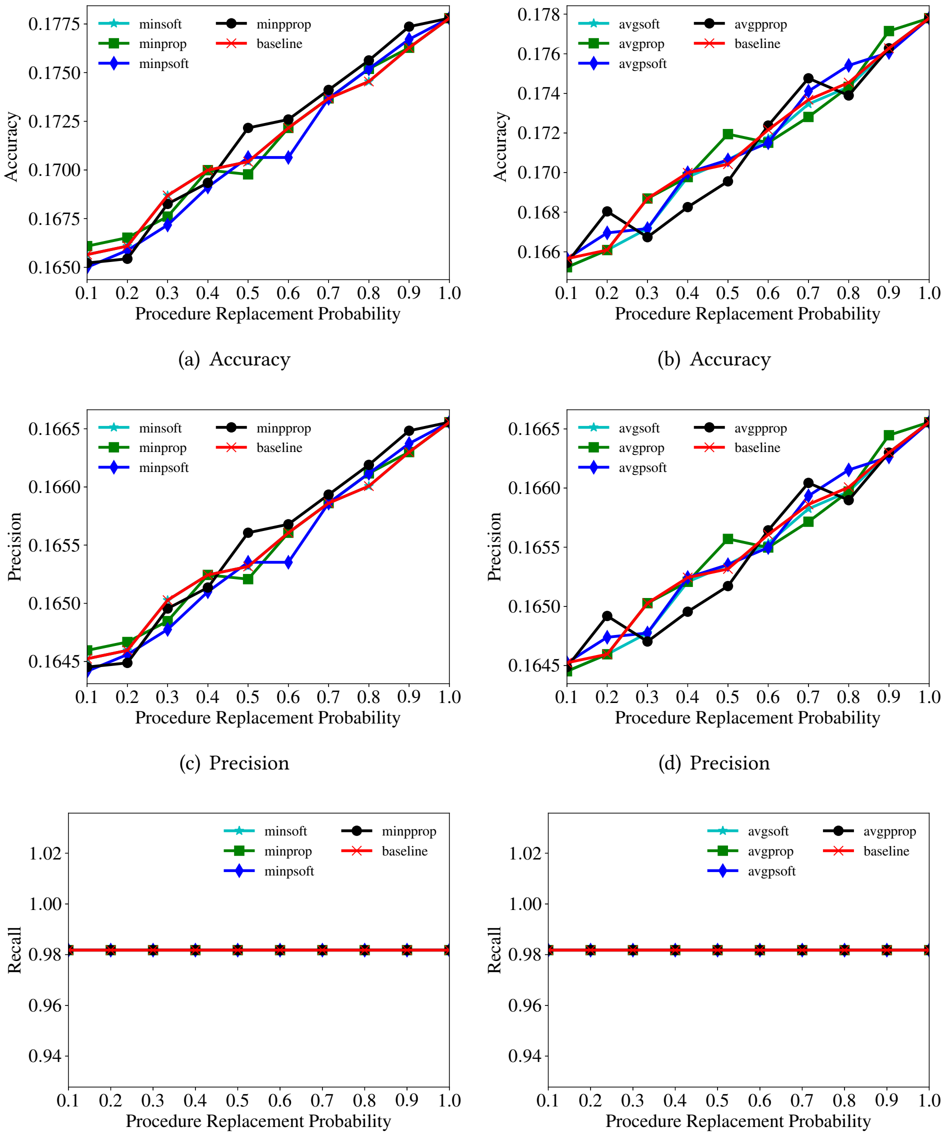
	Pooling Method	Procedure Probability Setting
minsoft	Minimum	Softmax
minpsoft	Minimum	Partitional softmax
minprop	Minimum	Proportions
minpprop	Minimum	Partitional proportions
avgsoft	Average	Softmax
avgpsoft	Average	Partitional softmax
avgprop	Average	Proportions
avgpprop	Average	Partitional proportions
baseline (uniform)	N/A	Uniform

selection method as a baseline negative claim generator. Note that, we use the legends from Table 3 for the two diagnosis distance pooling and four procedure probability settings in the evaluations.

In the following, we discuss the evaluation results for inpatient and outpatient datasets with respect to Skip-gram [30] based transformed features from diagnosis and procedure codes. We extract features of vector lengths 10, 20, and 50. We present the results based on vector length 20, as we observe similar results for the other vector lengths. In the following, we evaluate our approaches on both inpatient and outpatient datasets using one-class SVM. One-class SVM is applied on the positive claims to create a classification model and test on both positive and negative claims generated using our proposed approaches and the baseline approach. We use 80% of the positive claims for training and the remaining positive and negative claims as test data for evaluation using one-class SVM.

4.3.1 Comparative evaluation with uniform replacement as a baseline. Figure 7 presents the accuracy, precision, and recall scores on inpatient dataset using one-class SVM with minimum and average diagnosis distance pooling. One-class SVM is trained using only positive claims and tested on both true positive and synthetic negative claims. The results in Figures 7(a) through 7(d) show that the accuracy and precision scores increase linearly with negligible improvement with respect to procedure replacement probability thresholds. However, we observe significant false positive claims in the prediction results which contributes to high recall scores in Figure 7(e) and Figure 7(f). To demonstrate the cause of high recall on inpatient data, we provide a sample confusion matrix of one-class SVM classification in Table 4. The table shows a very high false positive score where minimum distance pooling with softmax procedure probability setting and 0.1 procedure replacement probability. The accuracy, precision, and recall scores from the confusion matrix are 0.162, 0.161, and 0.957, respectively. Therefore, the evidence supports the nature of one-class SVM which easily identifies positive claims. However, the classifier exhibits a very poor performance to identify negative claims. It mistakenly classifies 19,079 negative claims as positives. Therefore, the results suggest that our proposed distance pooling based transformations have generated negative claims which is very similar to the positive claims.

The primary reason for one-class SVM to perform poorly using all the proposed functions is that the test data includes negative claims generated from the diagnosis distance pooling approaches. Our approaches consist of a threshold parameter that regulates the contribution of positive procedures within a negative claim, which also creates different levels of difficulties for the classifiers. The poor classification performance of one-class SVM demonstrates that our negative claim generation techniques can simulate actual fraudulent claims which are similar to positive claims.



(a) Accuracy

(b) Accuracy

(c) Precision

(d) Precision

(e) Recall

(f) Recall

Fig. 7. Evaluation metrics of minimum (left) and average (right) poolings of negative claim generation and the baseline approaches on inpatient dataset using a one-class SVM classifier.

Figures 7(a), 7(c), and 7(e) present one-class SVM results with respect to minimum diagnosis distance pooling function. Minimum aggregation functions perform very similar to the baseline (uniform) approach that replaces positive procedures uniformly at random. However, we observe subtle differences in accuracy and precision scores between minimum and baseline (uniform) approaches. In Figure 7(a), most of our approaches demonstrate lower accuracy compared to the baseline (uniform) when procedure replacement probability is below 0.7. For the remaining

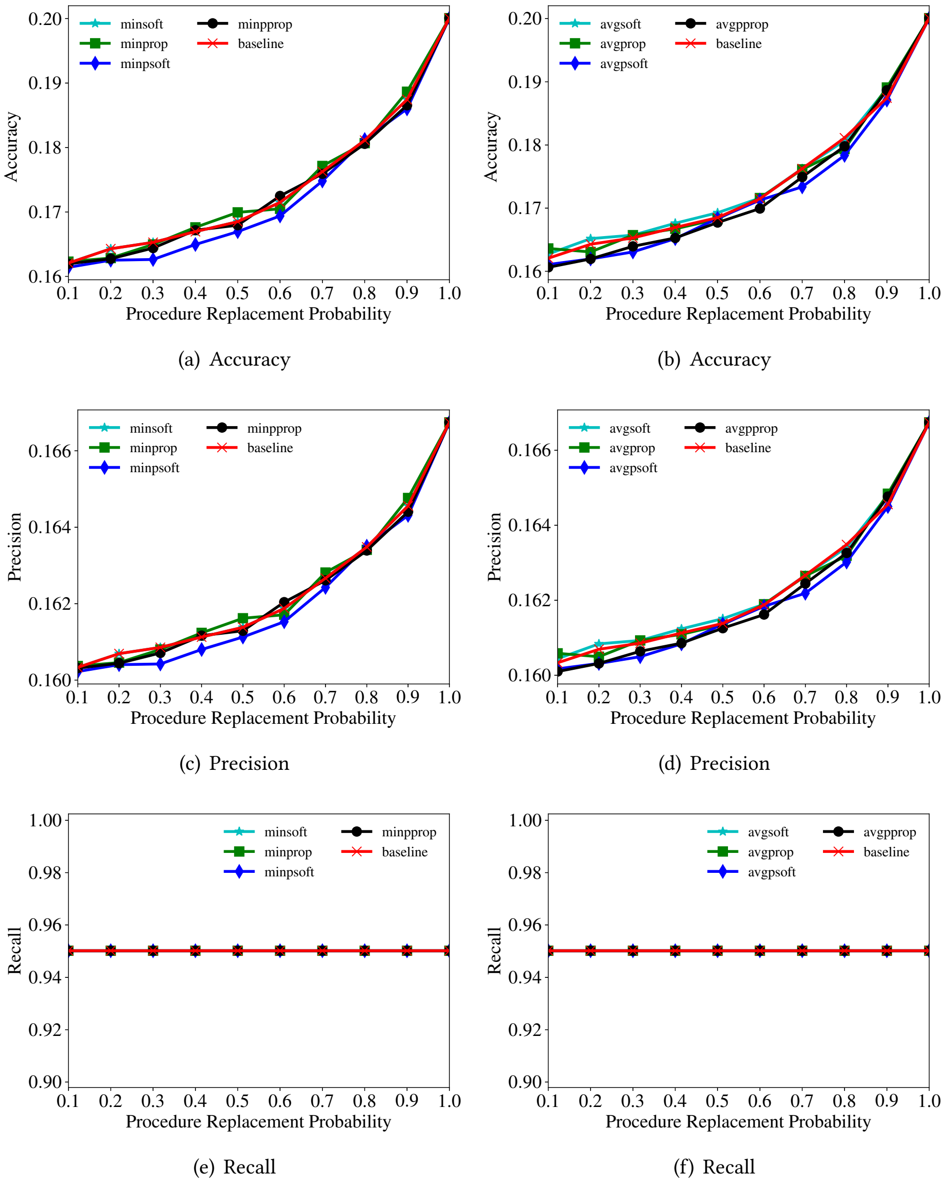
Table 4. Confusion matrix for minimum distance pooling with softmax procedure probability setting and 0.1 procedure replacement probability using one-class SVM classification on inpatient claims. *P* and *N* denote positive and negative classes, respectively.

		Predicted	
		P	N
Actual	P	3670	161
	N	19079	73

thresholds, our approaches either perform better or equal to the baseline. The reason behind this phenomena is that our methods generate negative claims which are highly similar to the positive claims in the vector space when replacement probability is lower. On the other hand, our methods generate relatively distinguishable claims with replacement probability of 0.7 and higher compared to the baseline. Precision scores in Figure 7(c) follows a similar pattern as accuracy presented in Figure 7(a). Figure 7(e) demonstrates the strength of one-class SVM irrespective of the negative claim generation method. The reason behind this is that the classifier is trained on a single type of claim data.

Figures 7(b), 7(d), and 7(f) present one-class SVM results with respect to average diagnosis distance pooling function on inpatient claims. Unlike minimum aggregation functions, our approaches with average aggregation demonstrate lower accuracy compared to the baseline (uniform) approach for higher replacement probability threshold of 0.8 or higher in Figure 7(b). It also shows that several of our approaches such as average-softmax performs poor for lower threshold probability compared to the baseline. Precision scores in Figure 7(d) follows a similar pattern as accuracy in Figure 7(b). The reason for the average aggregation method to generate negative claims closer to the positive claims within the inpatient claims is that it can choose a procedure in the negative claim which is closer to the directly connected procedure set. Finally, recall scores demonstrate one-class SVM's ability to recognize the claim types accurately regarding the trained dataset. It also exhibits the classifier's consistent performance with respect to different data generation approaches.

Next, we present the performance of the proposed approaches on the outpatient data using Figure 8. The figure shows that the classification results deteriorate steadily with the decrease in replacement probability threshold. The major reason for low probability threshold based claim features to be classified with lower performance is the graph structural differences between positive and negative claims. We also notice that the overall performance improves with respect to the regions of three threshold limits. The first two regions are from 0.1 to 0.5 and 0.5 to 0.8, which show a piece-wise linear increase. In the final region between 0.8 and 1.0 the metrics show a steep increase. Figures 8(a), 8(c), and 8(e) present accuracy, precision, and recall scores of our approaches based on minimum aggregation on outpatient dataset. The accuracy scores are lower for partitional softmax (*minpsoft*) and partitional proportions (*minpprop*) compared to the baseline (uniform) for procedure replacement probabilities between 0.1 and 0.7. For higher thresholds both *minpsoft* and *minpprop* either performs poorly (for threshold 0.9) or similarly. The softmax based procedure probability setting has limitation of being scale invariant which might not ensure proper emphasis on the procedures with shorter distances. On the other hand, proportions based procedure probability setting has limitation of providing majority of the total probability to the majority of the procedures with shorter distances compared to the few medium or longer distance procedures. To alleviate these issues, partitional proportions and partitional softmax based procedure probability settings are introduced. As a result, claims generated by both of these approaches perform poorly during the classification process. Due to similar reasons, we observe identical behavior in Figures 8(b), 8(d),



(a) Accuracy

(b) Accuracy

(c) Precision

(d) Precision

(e) Recall

(f) Recall

Fig. 8. Evaluation metrics of minimum (left) and average (right) poolings of negative claim generation and the baseline (uniform) approaches on outpatient dataset using a one-class SVM classifier.

and 8(f) for both partitional softmax and partitional proportions with respect to average procedure probability settings as well. We provide a sample confusion matrix of one-class SVM in Table 5 which shows a similar pattern compared to the results of inpatient claims in Table 4. In both cases, false positive claims are very high which contributes to higher recall shown in Figures 8(e) and 8(f).

It is important to note that our original outpatient claim data has a significant portion of the total claims with very low frequency of procedures. As a result, the replacement probability affects highly

Table 5. Confusion matrix for minimum distance pooling with softmax procedure probability setting and 0.1 procedure replacement probability using one-class SVM classification on outpatient claims. *P* and *N* denote positive and negative classes, respectively.

		Predicted	
		P	N
Actual	P	7086	403
	N	37348	94

for those claims with respect to the differences in procedure codes between positive and negative claims. The synthetically generated negative claims' partial bipartite structure looks completely different compared to the original positive claim. Note that, we may face situations where a negative claim is exactly similar to another negative claim. Those cases will reduce the number of generated negative claims in the dataset. We describe the expected negative data generation by replacing procedure codes in Figure 9 using a simple example. The illustration helps us to understand the impact of expected number of edge switchings between diagnoses and procedure codes on the bipartite graph structure for higher τ.

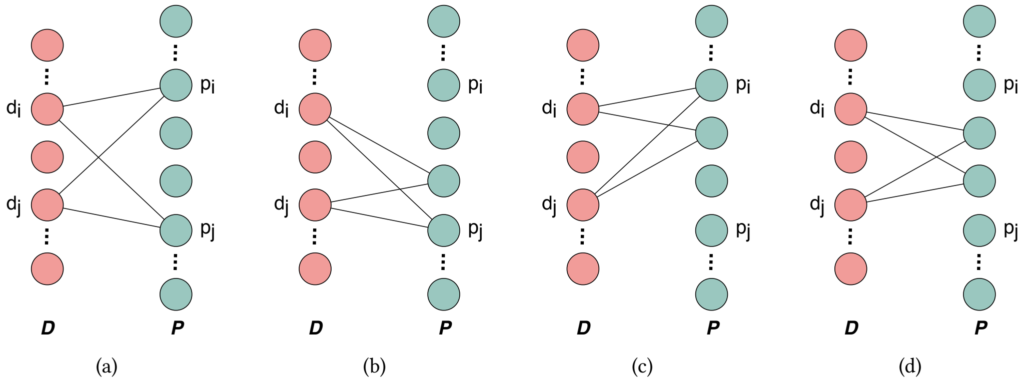
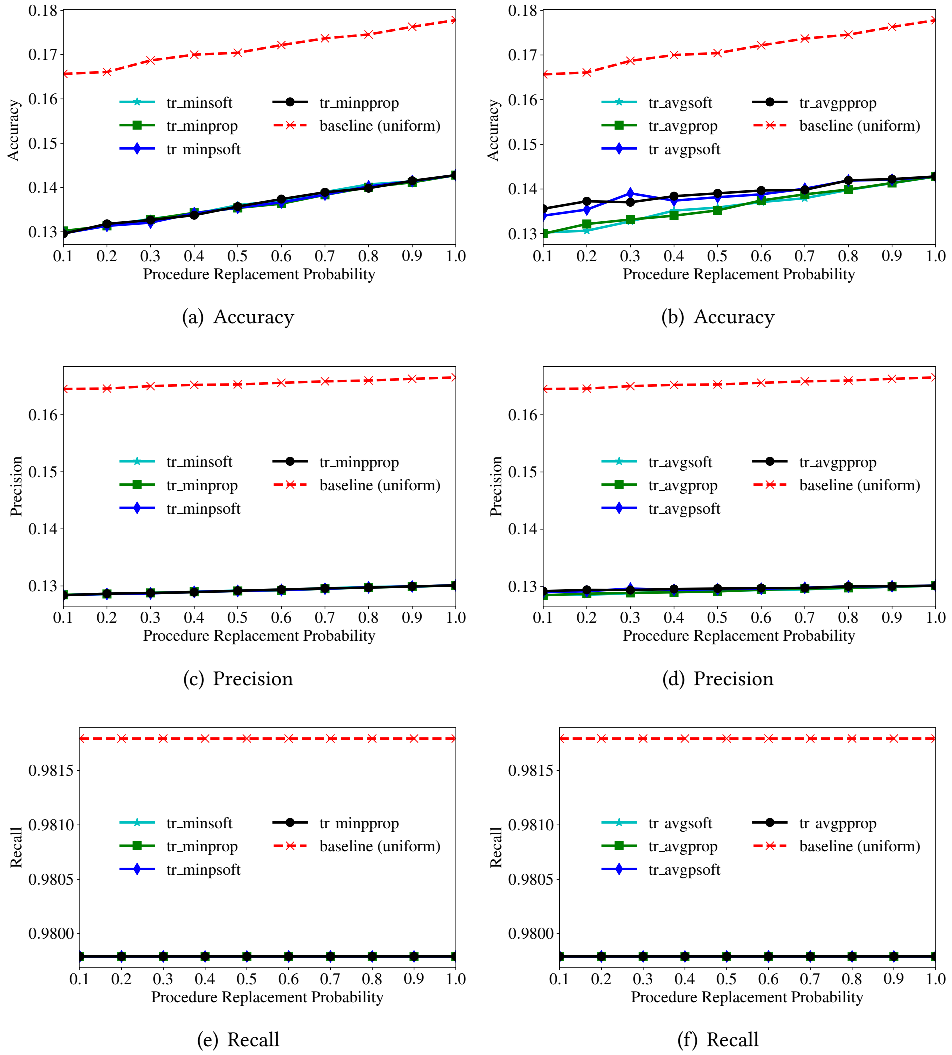


Fig. 9. Example illustrations of expected number of edge switchings between diagnoses and procedures when creating negative claims structures. (a) original positive claim, transformed claims with (b) $\tau = 0.5$, (c) $\tau = 0.5$, and (d) $\tau = 1.0$

Initially, we consider a positive claim with two diagnoses and two procedures in Figure 9(a). Next, we show three examples of expected number of edge switchings on the original partial bipartite graph structure of the positive claim. Figure 9(b) and 9(c) both are negative claim structures with $\tau = 0.5$ where edges connected with one procedure are unchanged. Figure 9(d), on the other hand is a complete edge switched negative claim. The illustration demonstrates that high probability thresholds are expected to create completely new structures in negative claims, which help classifiers to perform relatively better. We notice a similar behavior in Figure 7 as well.

The performance of one-class SVM is essentially poor for all procedure replacement probability thresholds, yet relatively better for higher thresholds due to different types of codes and graph structures created by our methods on both inpatient and outpatient datasets. This phenomena also denotes that one-class SVM can not differentiate between positive and negative class claims with varying levels of perturbations in the procedure codes.



(a) Accuracy

(b) Accuracy

(c) Precision

(d) Precision

(e) Recall

(f) Recall

Fig. 10. Evaluation metrics of minimum (left) and average (right) poolings of truncated negative claim generation and the baseline approaches on inpatient dataset using one-class SVM classifier.

4.3.2 Comparative evaluation with truncated procedure replacement. Figures 7 and 8 show that the baseline method behaves similar to the proposed approaches, especially at higher procedure replacement probabilities. As we analyzed our dataset further to explain this phenomena, we noticed that the dataset contains too many procedures at all distance levels. For example the inpatient dataset has only four distance levels, i.e., 1, 3, 5, and 7, covering 3126 procedure codes. Similarly, the outpatient dataset has only five distance levels, i.e., 1, 3, 5, 7, and 9, which covers 2831 procedure codes. Therefore, the assignment of different probabilities to different distance levels spreads out very thinly over all procedures. To empirically emphasize the difference between the proposed approaches and the baseline approach we introduce the modified versions of the proposed approaches, where the distant procedures are explicitly truncated. That is, the procedures that are located at five or more hops to the diagnoses group are assigned zero probabilities. To demonstrate this idea, we conducted a similar experiment over the inpatient dataset using one-class SVM with minimum and average diagnosis distance pooling. In this experiment we employed the truncated versions of the proposed approaches, i.e., truncated-minsoft, truncated-minprop, truncated-minpsoft, truncated-minpprop. Figure 10 presents the accuracy, precision and recall values of different procedure replacement probabilities with minimum and average distance pooling. In the figure the truncated versions of the proposed methods perform worse than the baseline approach in terms of all metrics for varying procedure replacement probabilities. Moreover, compared to Figure 7 the truncated versions perform worse than the original ones. That is, the truncated approaches generate more challenging negative claims for the classifier to classify which only indicates improved synthetic negative claims. Please note that a lower accuracy in our experiments implies better performing negative insurance claims. Table 6 shows the accuracy comparisons of the original and truncated versions of the proposed approaches with minimum and average poolings and 0.5 procedure replacement probability for the inpatient dataset. The ability of the classifier to distinguish between positive and negative claims decreases between 18.99% and 22.61% for all truncated approaches, where the maximum decrease is observed for the truncated proportional softmax. Clearly, replacing the positive procedures with the negative ones that are much closer to the diagnoses group creates a more challenging task for the classifier. In fact, we observed a similar behavior over the outpatient dataset as well. On the other hand, the baseline approach, i.e., uniform replacement at random, does not account for any distance levels by nature. Although a modified version of the baseline approach which truncates the distant procedures is possible, the modification tarnishes the concept of baseline, i.e., "uniform replacement at random".

Table 6. Accuracy comparisons of the original and truncated versions of the proposed approaches with minimum and average poolings and 0.5 procedure replacement probability for the inpatient dataset

		softmax	partitional softmax	proportions	proportional softmax
minimum	original	0.1704	0.1698	0.1706	0.1722
	truncated	0.1360 (20.18%↓)	0.1354 (20.26%↓)	0.1354 (20.63%↓)	0.1357 (21.20%↓)
average	original	0.1706	0.1719	0.1706	0.1796
	truncated	0.1359 (20.39%↓)	0.1352 (21.35%↓)	0.1382 (18.99%↓)	0.1390 (22.61%↓)

Finally, we conclude our experimental discussion by emphasizing the need for both truncated and non-truncated versions of procedure replacement techniques for generating synthetic negative claims. Based on our experimental results, discussions, and data formats we noticed that the non-truncated version of replacement is ideal for claim dataset where procedure code vocabulary has limited number of codes. The reason for this phenomena is that the small number of procedures have shorter group distance vector, which leads to selecting relevant procedures from corresponding positive claim. On the other hand, truncated version of replacement is ideal for claims dataset where procedure code vocabulary has large number of codes. The reason behind this is that the large number of procedures have longer group distance vector, which leads to selecting irrelevant procedures from corresponding positive claims. Therefore, we truncate the group distance vector to eliminate highly irrelevant procedures from corresponding positive claim. Please note that, inconsistent procedures are satisfactory within negative claims as long as they are consistent to their corresponding diagnoses in the diagnosis set. On the contrary, inconsistent diagnoses are natural in positive claims, as a patient may have multiple unrelated health conditions.

5 CONCLUSIONS

In this paper, we developed a procedure to generate negative insurance claims from a ground truth, positive health insurance claim dataset. In particular, we define positive claims as sets of diagnosis and procedure codes and exploit the relationship between them as bipartite graphs to quantify the relevancy or closeness between diagnosis groups and procedures. We presented two distance pooling and four procedure probability setting approaches and explored the variations of the negative insurance claim generation procedure. We used Medicare and Medicaid dataset from Centers for Medicare and Medicaid Services (CMS) and applied our approach to both inpatient and outpatient datasets which have discrepancies in their medical coding systems. We trained a one-class SVM classifier using the positive dataset and tested our approaches using both true positive and synthetic negative claims. The empirical results show that our procedure with average distance pooling, such as average softmax, performs worse or equal for all procedure replacement probabilities compared to the baseline method on the inpatient dataset. Note that a lower accuracy in our experiments implies better performing negative insurance claims. On the other hand, methods with minimum distance pooling performs poorly compared to the baseline method for procedure replacement probabilities of 0.7 or lower on the inpatient dataset. Regarding the outpatient claims dataset, minimum distance pooling with partitional softmax and partitional proportions exhibit lower accuracies compared to the baseline approach for procedure replacement probabilities of 0.7 or lower. We observe similar results for the methods based on average distance pooling as well. In addition, we introduced the truncated versions of the proposed approaches which reduce the accuracy of the classifier around 20%, on the average. In summary, our experimental results show that the generated negative claims are useful to simulate fraudulent claims in healthcare fraud research, where negative instances are not available. Moreover, the presented synthetic negative claim generation process is transferable to other domains where bipartite graph representations are meaningful.

REFERENCES

[1] Moustafa Alzantot, Supriyo Chakraborty, and Mani Srivastava. 2017. Sensegen: A deep learning architecture for synthetic sensor data generation. In *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE, 188–193.

[2] Emilie Lundin Barse, Hakan Kvarnstrom, and Erland Jonsson. 2003. Synthesizing test data for fraud detection systems. In *19th Annual Computer Security Applications Conference, 2003. Proceedings.* IEEE, 384–394.

[3] Richard Bauder, Raquel da Rosa, and Taghi Khoshgoftaar. 2018. Identifying medicare provider fraud with unsupervised machine learning. In *2018 IEEE International Conference on Information Reuse and Integration (IRI)*. IEEE, 285–292.

[4] Richard A Bauder and Taghi M Khoshgoftaar. 2018. The detection of medicare fraud using machine learning methods with excluded provider labels. In *The Thirty-First International Flairs Conference*.

[5] Christoph Baur, Shadi Albarqouni, and Nassir Navab. 2018. Generating highly realistic images of skin lesions with GANs. In *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*. Springer, 260–267.

[6] Elisa Bertino, Geeth de Mel, Alessandra Russo, Seraphin Calo, and Dinesh Verma. 2017. Community-based self generation of policies and processes for assets: Concepts and research directions. In *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2961–2969.

[7] Centers for Medicare and Medicaid Services. 2020. Research, Statistics, Data and Systems. https://www.cms.gov/Research-Statistics-Data-and-Systems/Downloadable-Public-Use-Files/SynPUFs/DE_Syn_PUF. Accessed January, 2020.

[8] Yunqiang Chen, Xiang Sean Zhou, and Thomas S Huang. 2001. One-class SVM for learning in image retrieval. In *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, Vol. 1. IEEE, 34–37.

[9] Wei Di and Melba M Crawford. 2011. View generation for multiview maximum disagreement based active learning for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* 50, 5 (2011), 1942–1954.

[10] Tahir Ekin, Luca Frigau, and Claudio Conversano. 2021. Health care fraud classifiers in practice. *Applied Stochastic Models in Business and Industry* (2021).

[11] Tahir Ekin, Francesca Ieva, Fabrizio Ruggeri, and Refik Soyer. 2018. Statistical medical fraud assessment: exposition to an emerging field. *International Statistical Review* 86, 3 (2018), 379–402.

[12] Tahir Ekin, Greg Lakomski, and Rasim Muzaffer Musal. 2019. An unsupervised Bayesian hierarchical method for medical fraud assessment. *Statistical Analysis and Data Mining: The ASA Data Science Journal* 12, 2 (2019), 116–124.

[13] FIND-A-CODE. 2020. Search for and lookup ICD 10 Codes, CPT Codes, HCPCS Codes, ICD 9 Codes, medical terms, medical newsletters, medicare documents and more. https://www.findacode.com/search/search.php. Accessed January, 2020.

[14] Font Awesome. 2020. Image Generated by Free Icons. https://fontawesome.com/license/free. Online.

[15] Maayan Frid-Adar, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. 2018. Synthetic data augmentation using GAN for improved liver lesion classification. In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*. IEEE, 289–293.

[16] Yongchang Gao, Chenfei Sun, Ruican Li, Qingzhong Li, Lizhen Cui, and Bin Gong. 2018. An efficient fraud identification method combining manifold learning and outliers detection in mobile healthcare services. *IEEE Access* 6 (2018), 60059–60068.

[17] Richard M Golden, Steven S Henley, Halbert White, and T Michael Kashner. 2019. Consequences of model misspecification for maximum likelihood estimation with missing data. *Econometrics* 7, 3 (2019), 37.

[18] Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. 2018. Long text generation via adversarial training with leaked information. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[19] Md Enamul Haque. 2020. *A Bipartite Graph Based Representation Learning for Healthcare Claims and Its Application to Fraudulent Claim Identification*. Ph.D. Dissertation. University of Louisiana at Lafayette.

[20] Md Enamul Haque and Mehmet Engin Tozal. 2021. Identifying Health Insurance Claim Frauds Using Mixture of Clinical Concepts. *IEEE Transactions on Services Computing* (2021).

[21] Joseph G Ibrahim, Haitao Chu, and Ming-Hui Chen. 2012. Missing data in clinical studies: issues and methods. *Journal of clinical oncology* 30, 26 (2012), 3297.

[22] ISIC. 2018. Skin Lesion Analysis Towards Melanoma Detection. https://challenge2018.isic-archive.com/. Accessed February, 2020.

[23] Xiao-Yuan Jing, Xinyu Zhang, Xiaoke Zhu, Fei Wu, Xinge You, Yang Gao, Shiguang Shan, and Jing-Yu Yang. 2019. Multiset feature learning for highly imbalanced data classification. *IEEE transactions on pattern analysis and machine intelligence* (2019).

[24] Donald B Johnson. 1977. Efficient algorithms for shortest paths in sparse networks. *Journal of the ACM (JACM)* 24, 1 (1977), 1–13.

[25] Saba Kareem, Rohiza Binti Ahmad, and Aliza Binit Sarlan. 2017. Framework for the identification of fraudulent health insurance claims using association rule mining. In *2017 IEEE Conference on Big Data and Analytics (ICBDA)*. IEEE, 99–104.

[26] Andreĭ Nikolaevich Kolmogorov and Albert T Bharucha-Reid. 2018. *Foundations of the theory of probability: Second English Edition*. Courier Dover Publications.

[27] Der-Chiang Li, Susan C Hu, Liang-Sian Lin, and Chun-Wu Yeh. 2017. Detecting representative data and generating synthetic samples to improve learning accuracy with imbalanced data sets. *PloS one* 12, 8 (2017), e0181853.

[28] Kevin Lin, Dianqi Li, Xiaodong He, Zhengyou Zhang, and Ming-Ting Sun. 2017. Adversarial ranking for language generation. In *Advances in Neural Information Processing Systems*. 3155–3165.

[29] Irum Matloob, Shoab Ahmed Khan, and Habib Ur Rahman. 2020. Sequence Mining and Prediction-Based Healthcare Fraud Detection Methodology. *IEEE Access* 8 (2020), 143256–143273.

[30] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[31] National Health Care Anti-Fraud Association. 2020. Consumer Info and Action. https://www.nhcaa.org/resources/health-care-anti-fraud-resources/consumer-info-action.aspx. Accessed January, 2020.

[32] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 311–318.

[33] Raphael Petegrosso, Zhuliu Li, Molly A Srour, Yousef Saad, Wei Zhang, and Rui Kuang. 2019. Scalable remote homology detection and fold recognition in massive protein networks. *Proteins: Structure, Function, and Bioinformatics* 87, 6 (2019), 478–491.

[34] Hong Liang Qiao. 2019. System and method of sentiment data generation. US Patent 10,198,506.

[35] Alyssa J Rolfe. 2021. Weighted risk models for dynamic healthcare fraud detection. *Risk Management and Insurance Review* (2021).

[36] Gokay Saldamli, Vamshi Reddy, Krishna S Bojja, Manjunatha K Gururaja, Yashaswi Doddaveerappa, and Loai Tawalbeh. 2020. Health Care Insurance Fraud Detection Using Blockchain. In *2020 Seventh International Conference on Software Defined Systems (SDS)*. IEEE, 145–152.

[37] Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural computation* 13, 7 (2001), 1443–1471.

[38] Lavanya Settipalli and GR Gangadharan. 2021. Healthcare fraud detection using primitive sub peer group analysis. *Concurrency and Computation: Practice and Experience* (2021), e6275.

[39] Yuliang Shi, Chenfei Sun, Qingzhong Li, Lizhen Cui, Han Yu, and Chunyan Miao. 2016. A fraud resilient medical insurance claim system. In *Thirtieth AAAI Conference on Artificial Intelligence*.

[40] Hoo-Chang Shin, Neil A Tenenholtz, Jameson K Rogers, Christopher G Schwarz, Matthew L Senjem, Jeffrey L Gunter, Katherine P Andriole, and Mark Michalski. 2018. Medical image synthesis for data augmentation and anonymization using generative adversarial networks. In *International workshop on simulation and synthesis in medical imaging*. Springer, 1–11.

[41] Hamilton O Smith, Clyde A Hutchison, Cynthia Pfannkoch, and J Craig Venter. 2003. Generating a synthetic genome by whole genome assembly: φX174 bacteriophage from synthetic oligonucleotides. *Proceedings of the National Academy of Sciences* 100, 26 (2003), 15440–15445.

[42] Jimeng Sun, Huiming Qu, Deepayan Chakrabarti, and Christos Faloutsos. 2005. Neighborhood formation and anomaly detection in bipartite graphs. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*. IEEE, 8–pp.

[43] László Szilágyi, Levente Kovács, and Sándor Miklós Szilágyi. 2014. Synthetic test data generation for hierarchical graph clustering methods. In *International Conference on Neural Information Processing*. Springer, 303–310.

[44] Jason Walonoski, Mark Kramer, Joseph Nichols, Andre Quina, Chris Moesel, Dylan Hall, Carlton Duffett, Kudakwashe Dube, Thomas Gallagher, and Scott McLachlan. 2018. Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *Journal of the American Medical Informatics Association* 25, 3 (2018), 230–238.

[45] Babak Zafari and Tahir Ekin. 2019. Topic modelling for medical prescription fraud and abuse detection. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 68, 3 (2019), 751–769.