CONAN: Complementary Pattern Augmentation for Rare Disease Detection

Limeng Cui,[1,2] **Siddharth Biswal,**[1,3] **Lucas M. Glass,**[1] **Greg Lever,**[1] **Jimeng Sun,**[3] **Cao Xiao**[1]
[1]Analytic Center of Excellence, IQVIA, Cambridge, MA, USA
[2]College of Information Sciences and Technology, The Pennsylvania State University, PA, USA
[3]College of Computing, Georgia Institute of Technology, Atlanta, GA, USA
lzc334@psu.edu, sbiswal7@gatech.edu, {lucas.glass, greg.lever, cao.xiao}@iqvia.com, jsun@cc.gatech.edu

Abstract

Rare diseases affect hundreds of millions of people worldwide but are hard to detect since they have extremely low prevalence rates (varying from 1/1,000 to 1/200,000 patients) and are massively underdiagnosed. How do we reliably detect rare diseases with such low prevalence rates? How to further leverage patients with possibly uncertain diagnosis to improve detection? In this paper, we propose a Complementary patter<u>n</u> Augmentatio<u>n</u> (CONAN) framework for rare disease detection. CONAN combines ideas from both adversarial training and max-margin classification. It first learns self-attentive and hierarchical embedding for patient pattern characterization. Then, we develop a complementary generative adversarial networks (GAN) model to generate candidate positive and negative samples from the uncertain patients by encouraging a max-margin between classes. In addition, CONAN has a disease detector that serves as the discriminator during the adversarial training for identifying rare diseases. We evaluated CONAN on two disease detection tasks. For low prevalence inflammatory bowel disease (IBD) detection, CONAN achieved .96 precision recall area under the curve (PR-AUC) and 50.1% relative improvement over the best baseline. For rare disease idiopathic pulmonary fibrosis (IPF) detection, CONAN achieves .22 PR-AUC with 41.3% relative improvement over the best baseline.

Introduction

There are more than 7,000 of diseases that are individually rare but collectively common. These rare diseases affect 350 million people worldwide and incur a huge loss in quality of life and large financial cost (Vickers 2019). As these diseases are rare individually, initial misdiagnosis is common. On average it can take more than seven years for rare disease patients to receive the accurate diagnosis with the help of 8 physicians (Shire 2013). Thus, it is important to detect and intervene with the rare disease before it becomes life-threatening and consumes excessive medical resources. In recent years, the availability of massive electronic health records (EHR) data enables the training of deep learning models for accurate predictive health (Xiao, Choi, and Sun 2018). However, current success mainly focuses on common chronic diseases such as Parkinson's disease progression modeling (Baytas et al. 2017; Che et al. 2017a) and heart failure prediction (Choi et al. 2018), deep learning models for rare disease prediction are lacking.

Two key challenges are presented for rare disease detection. First, the **low prevalence rates** of rare diseases limit the number of positive subjects in the training data (i.e., patients with a confirmed diagnosis of the rare disease). Thus the disease patterns are hard to extract. Second, there exist many **patients with uncertain diagnosis** due to the long period needed for rare diseases to be correctly diagnosed. The existence of large number of such uncertain patients can potentially help the disease detector perform better, as they are inherently close to positive patients who has confirmed diagnosis of the rare disease.

Related setting can be found in Positive-Unlabeled (PU) learning, which assumes the unlabeled data can contain both positive and negative examples (Bekker and Davis 2018). Existing PU learning methods (Elkan and Noto 2008; Kiryo et al. 2017) often identify reliable negative examples and then learn based on the labeled positives and reliable negatives (Liu et al. 2003). However, it is difficult to apply existing PU learning methods for the rare disease detection problem as the key difficulty here is to distinguish positive patients from negative ones with similar conditions. The reliable negative examples (e.g., healthy individuals and patients with similar diseases) will yield a more relaxed classification hyperplane and thus generate many false positive cases due to the low prevalence rates of rare diseases.

To mitigate the aforementioned problem, researchers adopt the generative adversarial networks (GANs) (Goodfellow et al. 2014) to augment the minor class and balance the distribution (McLachlan, Dube, and Gallagher 2016; Choi et al. 2017). However, several challenges are still present in GAN based methods:

1. They often focus on generating raw patient data, which is an extremely difficult problem on its own. The resulting synthetic data can easily be non-realistic and thus not useful for disease detection.
2. They often try to augment data based on the rare class only. However, due to the low prevalence rate, rare class

(rare disease patients) are insufficient to provide robust embedding that supports effective data augmentation.

3. Many GAN based methods often generate positive samples from Gaussian noises and apply a discriminator to distinguish real from fake data, which is not targeted toward rare disease detection.

To tackle these challenges, we propose *pattern augmentation* that can better preserve and enrich crucial patterns of the target disease. We also recognize that among negative subjects, there exist "borderline" cases that have uncertain diagnoses and potentially have the risk of rare diseases. For example, a rare disease idiopathic pulmonary fibrosis (IPF) shares compatible clinical, radiological, and pathological findings with a common chronic disease hypersensitive pneumonitis (Cordeiro, Alfaro, and Freitas 2013). Without any further investigation, no definite diagnosis could be made, leaving many patients to be *uncertain*.

Based on the above observations, we propose the Complementary pattern Augmentation (CONAN) framework which combines the idea of adversarial training and max margin classification for accurate rare disease detection. CONAN is enabled by the following technical contributions:

1. **Self-attentive and hierarchical embedding for better patient pattern characterization**. CONAN constructs an end-to-end hierarchical (visit- and patient-level) embedding model with two levels of self-attention to embed the raw patient EHR into latent pattern vectors. The resulting patterns can pay more attention to important codes and visits for each patient (Section. Self-attentive and Hierarchical Patient Embedding Net).
2. **Disease detection discriminator for improved pattern classification**. Unlike traditional GAN model that uses a discriminator to classify real or generated data, we construct a disease detector that serves as the discriminator during the adversarial training for identifying candidate positive samples by encouraging a max-margin between the two clusters in the generated complementary samples (Section. Disease Detector).
3. **Complementary GAN for boosted pattern augmentation**. CONAN uses an adversarial learning mechanism to use "uncertain" patients as seeds to generate complementary patient embedding for boosted pattern augmentation (Section. Complementary GAN).

We evaluated CONAN on two real-world disease detection tasks (Section. Experiments). The reported results show that for low prevalence inflammatory bowel disease detection, CONAN achieved 50.1% relative improvement in PR-AUC and 64.5% in F1 over best baseline medGAN. For rare disease idiopathic pulmonary fibrosis detection, CONAN has 41.3% relative improvement in PR-AUC and 39.3% in F1 over best baseline nnPU. An additional experiment shows CONAN performs well in early disease detection.

Related Work

Data Augmentation via GAN GAN consists of a generator that learns to generate new plausible but fake samples and a discriminator that aims to distinguish generated samples from real ones. The two networks are set up in minimax game, where the generator tries to fool the discriminator, and the discriminator aims to discriminate the generated samples (Goodfellow et al. 2014). GANs have shown superior performance in image generation, and also demonstrated initial success in generating synthetic patient data to address the data limitation. For example, the medGAN model generates synthetic EHR data by combining an autoencoder with GAN (Choi et al. 2017). While the ehrGAN model augments patient data in a semi-supervised manner (Che et al. 2017b). However, existing methods use the generator to fake samples, and once the discriminator is converged, it will not have high confidence in separating samples.

In this work, we devise a generator which can generate candidate positive and negative samples, and use them to enhance classification performances of the discriminator, rather than distinguishing fake data from real ones.

Deep Phenotyping The availability of massive EHR data enables training of complex deep learning models for accurate predictive health (Xiao, Choi, and Sun 2018). RETAIN (Choi et al. 2016) uses reverse time attention mechanism to detect influential past visits for heart failure prediction. T-LSTM (Baytas et al. 2017) handles irregular time intervals in the EHR data. Dipole (Ma et al. 2017) embeds the visits through a bidirectional GRU for diagnosis prediction. MiME (Choi et al. 2018) leverages auxiliary tasks to improve disease prediction under data insufficiency setting. Despite these achievements, existing works mainly focus on common chronic diseases, while deep learning models for rare disease prediction are lacking.

Positive-Unlabeled Learning In PU learning setting, positive samples are determined, while unlabeled samples can either be positive or negative. PU learning has attracted much attention in text classification (Liu et al. 2003), biomedical informatics (Claesen et al. 2015) and knowledge based completion (Galárraga et al. 2015). Two-step approaches (Zhou et al. 2004; Fung et al. 2005) first extract reliable negative and positive examples and then build the classifier upon them. Direct approaches (Elkan and Noto 2008; Sellamanickam, Garg, and Selvaraj 2011; Kiryo et al. 2017) treat unlabeled examples as negatives examples with class label noise and build the model directly. However, existing methods are not suitable for rare disease detection, as mentioned in the introduction. In this work, we exploit the uncertain samples as seeds to generate candidate positive and negative samples, and build the disease detector by encouraging a max-margin between the generated samples.

Method

Task Description

Definition 1 (Patient Records) In longitudinal EHR data, each patient can be represented as a sequence of multivariate observations: $\boldsymbol{P}_n = \{\mathcal{V}_n^{(t)}\}_{t=1}^{|\boldsymbol{P}_n|}$, where $n \in \{1, 2, \dots, N\}$, N is the total number of patients; $|\boldsymbol{P}_n|$ is the number of visits of the n-th patient. To reduce clutter, we will describe the algorithms for a single patient and drop the subscript

Table 1: List of basic symbols.

Symbol	Definition and description
$\boldsymbol{P}_n$	The EHR data of the n-th patient
$\mathcal{C}_s, \mathcal{C}_p$	Symptom and procedure code set
$\boldsymbol{c}_i^{(t)} \in \{0,1\}^{\lvert\mathcal{C}_*\rvert}$	i-th medical code in t-th visit
$\mathcal{V}^{(t)}$	Patient's t-th visit
$\mathbf{v}^{(t)}$	Patient's t-th visit embedding
$\mathbf{s}^{(t)}$	Annotation of patient's t-th visit
$\mathbf{u}^{(t)}$	Weight vector
$\alpha^{(t)}$	Normalized weight of t-th visit
$\mathbf{h}$	Patient embedding
$\mathbf{h}^+ \sim p_{\mathbb{R}^+}(\mathbf{h})$	Embeddings of positive patients
$\mathbf{h}^- \sim p_{\mathbb{R}^-}(\mathbf{h})$	Embeddings of negative patients
$y \in \{0,1\}$	Patient's disease label
$F(\boldsymbol{P}_n; \theta_e)$	Hierarchical embedding networks
$D(\mathbf{h}; \theta_c)$	Disease detector
$G(\mathbf{h}^-; \theta_g)$	Complementary embedding generator
$\hat{y} \in \{0,1\}$	Disease prediction of patient
$\mathbf{z} \sim p_{\hat{\mathbb{R}}}(\mathbf{h})$	Generated complementary embedding

n whenever it is unambiguous. For each patient, the visit $\mathcal{V}^{(t)} = \{\boldsymbol{c}_1^{(t)}, \ldots, \boldsymbol{c}_{|\mathcal{V}^{(t)}|}^{(t)}\}$ is a set of several symptom and procedure codes. For simplicity, we use $\boldsymbol{c}_i^{(t)}$ to indicate the unified definition for different type of medical codes; $|\mathcal{V}^{(t)}|$ is the number of medical codes. $\boldsymbol{c}_i^{(t)} \in \{0,1\}^{|\mathcal{C}_*|}$ is a multi-hot vector, where $\mathcal{C}_*$ denotes the symptom code set and the procedure code set, and $|\mathcal{C}_*|$ is the size of the code set.

The patient representation/embedding is denoted as $\mathbf{h}$. Assume we have M positive embeddings $\mathbf{h}^+ \sim p_{\mathbb{R}^+}(\mathbf{h})$, and $N - M$ negative embeddings $\mathbf{h}^- \sim p_{\mathbb{R}^-}(\mathbf{h})$, where $\mathbb{R}^+$ and $\mathbb{R}^-$ represents the positive samples' and negative samples' space respectively. $M \ll N$ in extremely imbalanced cases. Table 1 lists notations used throughout the paper.

Problem 1 (Rare Disease Detection) Given each patient $\boldsymbol{P}_n$, we want to learn self-attentive and hierarchical patient embedding net $F : F(\boldsymbol{P}_n; \theta_e) \to \mathbf{h}$ to get patient embeddings which becomes the input with a rare disease detection function to determine if the patient has rare disease $D : D(\mathbf{h}; \theta_c) \to \hat{y} \in \{0,1\}$.

Problem 2 (Complementary Pattern Augmentation) Given a set of negative patients' visit embeddings $\mathbf{h}^-$, we want to learn a generator G that can generate complementary embeddings: $G : G(\mathbf{h}^-; \theta_g) \to \mathbf{z}$.

The CONAN Framework

As illustrated in Fig. 1, CONAN includes the following components: self-attentive and hierarchical patient embedding net, complementary GAN, and a disease detector component. Next, we will first introduce these modules and then provide details of training and inference of CONAN.

Self-attentive and Hierarchical Patient Embedding Net Motivated by previous work (Ma et al. 2017), we leverage the inherent multilevel structure of EHR data to learn patient embedding. The hierarchical structure of EHR data begins with the patient, followed by visits the patient experiences, then the set of diagnosis codes, procedure, and medication codes recorded for that visit.

1. *Visit Embedding.* Given a patient's visit with codes $\mathcal{V}^{(t)} = \{c_i^{(t)}\}_{i=1}^{|\mathcal{V}^{(t)}|}$, we first embed the codes to vectors via a multi-layer Transformer (Vaswani et al. 2017), which is a multi-head attention based architecture. It takes the code embeddings as input and derives code embedding $\mathbf{v}^{(t)}$ for a patient at t-th visit. As the medical codes in one visit is not ordered, it is proper to use Transformer as it relies entirely on self-attention to compute representations of its input and output without using sequence aligned RNNs or convolution.
   Specifically, we remove the position embedding in the Transformer, and get the patient representation of t-th visit is computed as follows:
   $$\mathbf{v}^{(t)} = \mathrm{Transformer}(\mathcal{V}^{(t)}) \quad (1)$$
2. *Patient Embedding.* To capture the patient embedding across multiple hospital visits, we use bidirectional LSTM (Bi-LSTM) as an encoder. An obvious limitation of LSTM is that it can only use previous context. The Bi-LSTM overcomes this gap by incorporating both directions with forward layer and backward layer. To be specific, given visit embedding $\mathbf{v}^{(t)}$ of a patient, the patient embedding is computed as below.
   $$\begin{aligned}\overrightarrow{\mathbf{s}}^{(t)} &= \mathrm{LSTM}(\overrightarrow{\mathbf{s}}^{(t-1)}, \mathbf{v}^{(t)})\\ \overleftarrow{\mathbf{s}}^{(t)} &= \mathrm{LSTM}(\overleftarrow{\mathbf{s}}^{(t+1)}, \mathbf{v}^{(t)})\end{aligned} \quad (2)$$
   We obtain an annotation for visit $\mathbf{v}^{(t)}$ by concatenating the forward hidden state $\overrightarrow{\mathbf{s}}^{(t)}$ and the backward hidden state $\overleftarrow{\mathbf{s}}^{(t)}$, $\mathbf{s}^{(t)} = [\overrightarrow{\mathbf{s}}^{(t)}, \overleftarrow{\mathbf{s}}^{(t)}]$, which summarizes the information of the all the visits centered around $\mathbf{v}^{(t)}$.
   In addition, to find out which visit is important in the whole course, we use self-attention mechanism again to measure the importance of each visit:
   $$\begin{aligned}\mathbf{u}^{(t)} &= \mathbf{u}_s^\top \tanh(\mathbf{W}_s \mathbf{s}^{(t)} + \mathbf{b}_s)\\ \alpha^{(t)} &= \frac{\exp(\mathbf{u}^{(t)})}{\sum_t \exp(\mathbf{u}^{(t)})}\\ \mathbf{h} &= \sum_t \alpha^{(t)} \mathbf{s}^{(t)}\end{aligned} \quad (3)$$
   where $\mathbf{u}^{(t)}$ is a representation of $\mathbf{s}^{(t)}$, weight vector $\mathbf{u}_s$ is randomly initialized and learned through the training process, $\alpha^{(t)}$ is the normalized weight of the t-th visit, and $\mathbf{h}$ is the embedding of a patient which summarizes all the visits and their importance.

The above two-level embedding process of patient's visits is denoted as self-attentive and hierarchical patient embedding net $\mathbf{h} = F(\boldsymbol{P}_n; \theta_e)$, where θ_e represents all the parameters to be learned for simplicity.

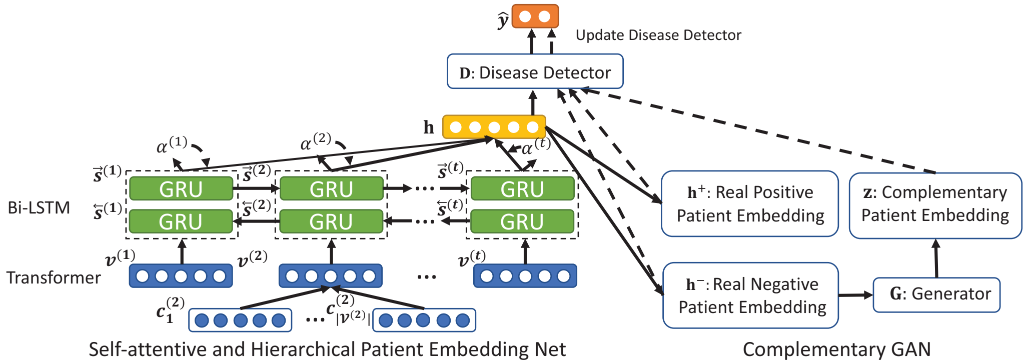


Figure 1: Framework overview. It contains three main parts: self-attentive and hierarchical patient embedding net, disease detector and complementary generator. Firstly, we compute the patient's embedding by Eqn. 1, 2 and 3 through a hierarchical attention mechanism. Then, we use focal loss to train a disease detector. Finally, we set up a complementary generator (Eqn. 4) to convert negative embeddings to positive with slight modifications, and fine-tune the disease detector with the real and generated embeddings (Eqn. 7).

Complementary GAN As discussed earlier, some negative samples can be converted to positive samples (candidate positive samples) with slight modifications, while others are still negative samples (candidate negative samples). From an adversarial perspective, the disease detector should be "smart" enough to classify the candidate positive samples as positive, and others as negative. We denote the candidate positive and negative samples generated as "complementary samples", and use them to help enhance the performance of the rare disease detector.

In this section, we introduce a complementary GAN algorithm to generate complementary embedding by using the negative embeddings $\mathbf{z} = G(\mathbf{h}^-; \theta_g) \sim p_{\hat{\mathbb{R}}}(\mathbf{h})$, where $\hat{\mathbb{R}}$ is the distribution space of the complementary samples. The learning purpose of the complementary GAN is to balance the two distribution $p_{\mathbb{R}^+}(\mathbf{h})$ and $p_{\hat{\mathbb{R}}}(\mathbf{h})$. The generator intends to assign all the generated samples to the positive class with the disease detection $D(\mathbf{h}; \theta_c)$. Therefore, the loss function is designed as follows:

$$\begin{aligned}\mathcal{L}_g(\theta_g) = &- \lambda \mathbb{E}_{\mathbf{h}^- \sim p_{\mathbb{R}^-}}[\log D(G(\mathbf{h}^-; \theta_g); \hat{\theta}_c)] \\ &+ ||G(\mathbf{h}^-; \theta_g) - \mathbf{h}^-||_2\end{aligned} \tag{4}$$

where the first term measures the difference between the output probability and the positive distribution, and the second term measures the change between the original negative samples and the converted samples. λ is a trade-off parameter that controls the relative importance of the two terms. We tried $\lambda = \{0.01, 0.05, 0.1, 0.5, 1\}$ and $\lambda = 0.05$ works best.

Disease Detector The disease detector can classify both generated and original embedding to learn important codes/visits. It is built on top of the hierarchical embedding, taking the patient representation $\mathbf{h}$ as input. We denote the disease detector as $D(\cdot; \theta_c)$, where θ_c represents the parameters to be learned. The output of the disease detector is the probability of this patient being positive.

The goal of the disease detector if to determine whether a patient has the disease or not. In rare disease detection problem, the discriminator evaluate $10^4 - 10^5$ patients but only a few have the target rare disease. Such a data imbalance issue causes two learning problems: (1) the training is insufficient as the easy negatives do not contain much information; (2) the easy negatives might degenerate the model. To efficiently train on all examples, we employ focal loss (Lin et al. 2017):

$$\begin{aligned}\mathcal{L}_c(\theta_e, \theta_c) = &-\mathbb{E}_{\mathbf{h}^+ \sim p_{\mathbb{R}^+}}[\alpha(1 - D(\mathbf{h}; \theta_c))^\gamma \log D(\mathbf{h}; \theta_c)] \\ &- \mathbb{E}_{\mathbf{h}^- \sim p_{\mathbb{R}^-}}[(1 - \alpha) D(\mathbf{h}; \theta_c)^\gamma \log(1 - D(\mathbf{h}; \theta_c))]\end{aligned} \tag{5}$$

where γ is a focusing parameter, which focuses more on hard and easily mis-classified examples, and α is the weight assigned to the rare class. $\gamma = 2$ and $\alpha = 0.25$ works best based on the rule of thumb (Lin et al. 2017).

The disease detector is trained as follows. First, the disease detector works with the embedding net to get patient representation $\mathbf{h}$. Second, with all the negative samples that are converted as positive through the generator, we want the detector to distinguish the candidate positive samples from the candidate negative samples, through maximizing the distance between two classes. Thus, we use the conditional entropy of labelings (Dai and Hu 2010; Deng et al. 2017) to maximize the margin:

$$\begin{aligned}H(\theta_c) = &- D(\mathbf{z}; \theta_c) \log(D(\mathbf{z}; \theta_c)) \\ &- (1 - D(\mathbf{z}; \theta_c)) \log(1 - D(\mathbf{z}; \theta_c))\end{aligned} \tag{6}$$

So we combine this goal with focal loss to form the final loss of the detector to further fine-tuning the parameter θ_c:

$$\mathcal{L}_s(\theta_c) = \mathcal{L}_c(\hat{\theta}_e, \theta_c) + \eta H(\theta_c) \tag{7}$$

where η is a weighting factor. We set $\eta = 0.05$.

Training and Inference with CONAN

During the training stage, in the first stage, the self-attentive and hierarchical patient embedding net $F(\cdot; \theta_e)$ work with disease detector $D(\cdot; \theta_c)$ to minimize the detection loss $\mathcal{L}_c(\theta_e, \theta_c)$, so as to provide insight into which codes and visits contribute to the rare disease. In the second stage, the generator $G(\cdot;\theta_g)$ tries to fool the detector $D(\cdot;\theta_c)$ by minimizing the loss $\mathcal{L}_g(\theta_g)$. Also by minimizing the discrimination loss $\mathcal{L}_s(\theta_c)$, the detector $D(\cdot;\theta_c)$ not only tries to discriminate positive embeddings from negative embeddings, but also maximize the margin between the two clusters of generated samples. The detailed training steps are summarized in Algorithm 1.

Algorithm 1: CONAN for Rare Disease Detection.

```
Input: Training set, training epochs for self-attentive
       and hierarchical patient embedding net $N_e$ and
       complementary GAN $N_g$
Output: Well-trained rare disease detector and
        complementary patient data embedding $\mathbf{z}$
for $i = 1, \ldots, N_e$ do
    Minimize the detection loss $\mathcal{L}_c(\theta_e, \theta_c)$ in Eq. 5;
end
foreach patient in dataset do
    Compute patient's embedding $\mathbf{h}$ by Eq. 1, 2 and 3;
end
for $i = 1, \ldots, N_g$ do
    Minimize the generation loss $\mathcal{L}_g(\theta_g)$ in Eq. 4 with
      negative patient's embedding $\mathbf{h}^-$;
    Compute the generated samples $\mathbf{z}$;
    Feed the detector with both the original patient's
      embedding $\mathbf{h}$ and generated sample $\mathbf{z}$;
    Minimize the final detection loss $\mathcal{L}_s(\theta_c)$ in Eq. 7;
end
```

Experiments

Experimental Setup

Data We leverage data from IQVIA longitudinal prescription (Rx) and medical claims (Dx) databases, which include hundreds of millions patients' clinical records. In our study, we focus on one rare disease and one low prevalence disease.

1. **Idiopathic Pulmonary Fibrosis (IPF)** is a pulmonary disease that is characterized by the formation of scar tissue within the lungs in the absence of any known provocation (Meltzer and Noble 2008). It is a rare disease which affects approximately 5 million persons worldwide, with prevalence rate at 0.04%
2. **Inflammatory Bowel Disease (IBD)** is a broad term that describes conditions characterized by chronic inflammation of the gastrointestinal tract. The two most common inflammatory bowel diseases are ulcerative colitis and Crohn's disease. IBD has a low prevalence. Overall, the prevalence of IBD is 439 cases per 100,000 persons.

For both datasets, we extracted patient records, including medical/diagnosis/procedure codes at visit level from January 2010 to April 2019, which include total 168,514 distinct medical codes. Each visit contains patient id, time of the visit, one symptom code and one diagnose/procedure. Data statistics are provided in Table 2.

Table 2: Statistics of datasets. The disease prevalence rates are the same as case/control ratio in test set.

	IPF	IBD
Category	rare	low prevalence
Prevalence	0.04%	0.44%
Positive	9,996	1,405
Negative	24,757,572	108,047
Ave. # of visit	597.36	798.25

Baselines We consider the following baseline methods:

- **LR**: We first embed each code into a vector, then concatenate all the vectors together, and feed it to the model.
- **PU-SVM** (Elkan and Noto 2008): PU-SVM is a well-known PU learning model. It labels positive training examples at random. The data are processed similarly to LR.
- **nnPU** (Kiryo et al. 2017): A PU learning model that is more robust against overfitting, and allows for using deep neural networks given limited positive data. The data are processed similarly to LR.
- **RNN**: We feed the code embeddings to a fully-connected RNN. The output generated by the RNN is directly used to predict the rare disease.
- **T-LSTM** (Baytas et al. 2017): T-LSTM handles time irregularity. Similar to RNN, we feed the embeddings to T-LSTM model, and use the output to predict the disease.
- **SMOTE** (Chawla et al. 2002): SMOTE is an oversampling method which randomly replicates minority class examples. We use LR and RNN to generate embeddings, and then use SMOTE to augment the positive class.
- **RETAIN** (Choi et al. 2016): RETAIN is a two-level attention-based neural model which detects significant past visits and clinical variables. It receives the EHR data in a reverse time to mimic the practical clinical course.
- **medGAN** (Choi et al. 2017): medGAN generates synthetic EHR data. We generate minority class EHR data and feed the output of the encoder into an MLP with cross-entropy loss to predict disease.
- **SSL GAN** (Yu et al. 2019): SSL GAN can augment positive embeddings, and facilitate rare disease detection by leveraging both positive and negative samples.
- **Dipole** (Ma et al. 2017): Dipole employs bidirectional recurrent neural networks to embed the EHR data. It introduces the attention mechanism to measure the relationships of different visits for the diagnosis prediction.

Metrics The following three metrics are used:

1. **Area Under the Precision-Recall Curve** (PR-AUC):

$$\text{PR-AUC} = \sum_{k=1}^{n} \text{Prec}(k)\Delta\text{Rec}(k),$$

where k is the k-th precision and recall operating point $(\text{Prec}(k), \text{Rec}(k))$.

2. **F1 Score**: $\text{F1 Score} = 2 \cdot (\text{Prec} \cdot \text{Rec})/(\text{Prec} + \text{Rec})$, where Prec is precision and Rec is recall.
3. **Cohen's Kappa**: $\kappa = (p_o - p_e)/(1 - p_e)$, where p_o is the observed agreement (identical to accuracy), and p_e is the expected agreement, which is probabilities of randomly seeing each category.

Implementation Details We implement all models with Keras [1]. We sample two imbalanced training sets for each dataset, with a ratio of 10% and 1% for positive samples. For the testing set, we extract the data using the actual disease prevalence rate shown in Table 2. We set 128 for dimensions of patient embedding. For the complementary GAN, the disease detector serves as the discriminator, and the complementary generator has two hidden layers with 128 dimensions. The output layer of the generator has the same dimension as the patient embedding. The training epoch of complementary GAN is 1000. For all models, we use RMSProp (Hinton, Srivastava, and Swersky 2012) with a minibatch of 512 patients, and the training epoch is 30. In order to have a fair comparison, we use focal loss (with $\gamma = 2$ and $\alpha = 0.25$) and set the output dimension as 128 for all models. The vocabulary size is consistent with ICD-9 diagnosis codes. The sequence length is chosen according to the average number of visit per patient in Table 2. For RNN and LSTM, the hidden dimensions of the embedding layer are set as 128. For other methods, we follow the network architectures in the papers. All methods are trained on an Ubuntu 16.04 with 128GB memory and Nvidia Tesla P100 GPU.

Results

Performance Comparison For the training data, we tried positive sample ratios, 10% and 1%, and report better results of two ratios on IBD and IPF dataset in Table 3. The best results are presented in bold figures. For IBD, the methods perform better by using 10% positive samples. For IPF, the methods perform better with 1% positive samples. We assume that the method performs best when the ratio of positive in training data is close to the ratio in test data, which is the actual prevalence rate of that disease. We also tried 0.1% and 50% positive on both datasets but did not get satisfying results. For 0.1% positive, all the samples are classified into the negative class regardless of the combinations of hyperparameters. The medical knowledge about negative patients suggests us that they can not be strictly counted as a class, so they contribute little to the identification of positive patients. For 50% positive, the false positive rate is high on the test data, which indicates that this training strategy is not practical for the real-world problem.

From Table 3 we can observe that the performance of LR, RNN and T-LSTM are less satisfactory, due to the complexity of the disease progression during long clinical courses.

Among the hierarchical embedding methods, RETAIN and Dipole are two hierarchical embedding methods with an attention mechanism. We can observe that Dipole performs better than RETAIN. For RETAIN, it models the EHR data in a reverse time to mimic the practical clinical course. The recent visits receive greater attention than previous visits. It is not very practical for the rare disease. Due to the complexity of the rare disease, the diagnoses/symptoms of similar diseases may intertwine together during the disease progression.

Regarding PU learning methods, including PU-SVM and nnPU, they are designed to exploit the reliable negative cases in the unlabeled data. However, these methods may not be suitable for the rare disease detection problem. As the main challenge in rare disease detection is to distinguish the patients with the rare disease (e.g., IPF) to patients with similar diseases (e.g., hypersensitivity pneumonitis), but the detected reliable negative cases are healthy people or patients with irrelevant diseases, which contribute little to the detection.

Oversampling (SMOTE_{LR} and SMOTE_{RNN}) and generative (medGAN and SSL GAN) models perform better than simple sequential models in most cases, which shows the effectiveness of the data augmentation.

Ablation Study We conduct an ablation study to understand the contribution of each component in CONAN. We remove/change the GAN module as below. The parameters in all the variants are determined with cross-validation, and the best performances are reported in Table 4.

- w/o GAN: w/o GAN is a variant of CONAN, which only contains the hierarchical patient embedding net.
- w GAN: w GAN is a variant of CONAN, which replaces the complementary GAN with the regular GAN, which is used to augment the positive embeddings.
- w cGAN: this method is CONAN, which incorporates the complementary GAN.

The results indicate that, when we solely use the self-attentive and hierarchical patient embedding net, the performances are largely reduced. It suggests the necessity of data augmentation. When we use the regular GAN to augment the data, performance are improved but still lower than CONAN. By augmenting the positive samples, the disease detector will have more confidence in detecting the positive samples. In other words, it detects more samples as positive, which yields a much higher false positive rate but doesn't decrease the false negative rate much as the ratio of positive samples is extremely low.

Through the ablation study of CONAN, we conclude that (1) data augmentation can contribute to the low prevalence rate disease detection performance; (2) complementary GAN is necessary for rare disease detection.

Early Disease Prediction We test how CONAN performs on early disease prediction for unseen patients. We select the first $x\%$ of visits in each patient's records for testing, where x is varied as $\{100, 50, 20\}$. Table 5 shows comparison results in terms of the PR AUC, F1 Score and Cohen's Kappa of early disease prediction. The CONAN achieves a satisfactory performance when $x = 50$ and $x = 20$, and it is still competitive compared with the baselines. It indicates that once we train the model, for unseen patients, we can predict their conditions at an early stage.

[1] https://github.com/cuilimeng/CONAN

Table 3: Performance Comparison on **IPF (rare disease, prevalence rate** 0.04%) and **IBD (low prevalence disease, prevalence rate** 0.44%) datasets. CONAN outperforms all state-of-the-art baselines including GAN based and PU learning baselines.

Dataset	Metric	LR	PU-SVM	nnPU	RNN	T-LSTM	SMOTE$_{LR}$	SMOTE$_{RNN}$	RETAIN	Dipole	SSL GAN	medGAN	CONAN
IBD	PR-AUC	0.2765	0.5321	0.5682	0.4373	0.2241	0.3464	0.4471	0.3135	0.5417	0.6072	0.6385	**0.9584**
	F1 Score	0.3651	0.4982	0.4392	0.4332	0.3016	0.4341	0.4642	0.3594	0.5528	0.5416	0.5834	**0.9601**
	Cohen's Kappa	0.3249	0.5123	0.4624	0.4440	0.2886	0.3451	0.4895	0.3106	0.5904	0.5453	0.4851	**0.9595**
IPF	PR-AUC	0.0798	0.1141	0.1578	0.0090	0.0084	0.0406	0.0187	0.1016	0.1183	0.0206	0.0954	**0.2229**
	F1 Score	0.1529	0.0915	0.1682	0.0169	0.0211	0.0673	0.0293	0.1345	0.0969	0.0272	0.0729	**0.2343**
	Cohen's Kappa	0.1369	0.0835	0.1397	0.0261	0.0752	0.0208	0.0429	0.1470	0.1060	0.0372	0.0612	**0.2339**

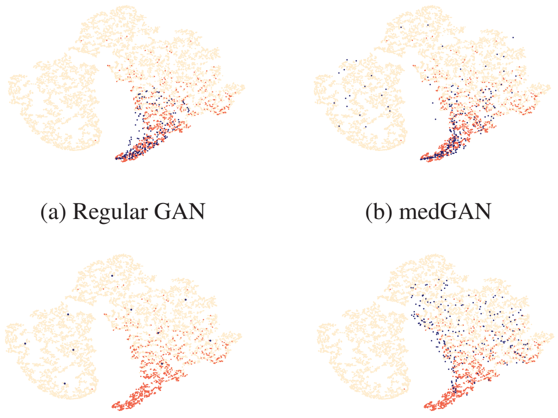
Table 4: Abalation study of CONAN demonstrated the advantage of complementary pattern augmentation.

Dataset	Metric	w/o GAN	w GAN	CONAN
IBD	PR AUC	0.8097	0.9323	0.9584
	F1 Score	0.8386	0.9566	0.9601
	Cohen's Kappa	0.8590	0.9560	0.9595
IPF	PR AUC	0.1796	0.2023	0.2229
	F1 Score	0.1119	0.1768	0.2343
	Cohen's Kappa	0.0767	0.1762	0.2339

Table 5: The results on early prediction indicated that we can use CONAN to predict patients' conditions at an early stage.

Dataset	Metric	% Visits in Test Data		
		100%	50%	20%
IBD	PR AUC	0.9584	0.9474	0.7313
	F1 Score	0.9601	0.9531	0.7993
	Cohen's Kappa	0.9595	0.9473	0.7629
IPF	PR AUC	0.2229	0.2105	0.0843
	F1 Score	0.2343	0.2262	0.1024
	Cohen's Kappa	0.2119	0.2056	0.0758

Visualize Generated Embedding We project the three types of patient's embeddings of IPF dataset, including real positive, real negative and generated, to a two-dimensional space by t-SNE (Maaten and Hinton 2008) and show the projection in Fig. 2. The red dots indicate the positive embeddings. The almond dots indicate the negative embeddings. The navy dots in four subfigures indicate the samples generated by regular GAN, medGAN, a PU learning method (Liu et al. 2003), and CONAN respectively. The generated samples of the four methods are distributed differently. The samples generated by regular GAN have the same distribution with the real positive samples. As a result, it would give the detector more confidence to classify the "borderline" patients into positive, which may yield a high false positive rate. We use medGAN to generate both positive and negative cases to balance the overall class distribution, but the same issue as regular GAN is still unsolved. The reliable negative samples generated by the PU learning method span the negative samples' space, which contribute little to the rare disease detection. The complementary samples generated by CONAN lie in between of positive and negative samples, which can help the disease detector update its hyperplane by encouraging a max-margin between the generated samples.



(a) Regular GAN

(b) medGAN

(c) PU Learning

(d) CONAN

Figure 2: 2D visualization of patient embeddings of IPF data: real negative (almond), real positive (red), and generated (navy).

Conclusion

In this paper, we proposed CONAN, a pattern augmentation method for rare and low prevalence disease detection. CONAN uses the embedding of negative samples as seeds to generate complementary patterns with a complementary GAN. The generator can convert the negative embedding to fool the discriminator, while the disease detector serves as the discriminator to distinguish the positive and negative samples by maximizing a margin between the generated samples. After the training, the discriminator can be used for detecting positive patients. Experiments on real-world datasets demonstrated the strong performance of CONAN.

The CONAN method can also be extended to other application domains for classification problems with imbalanced data, such as fraud detection and recommendation. Besides, there are several interesting future directions that need investigations. First, we can incorporate the data from similar diseases to guide the generation process and obtain more distinguishable patient embeddings. Second, other data sources, such as the doctor notes can be considered for better embedding. Third, time intervals between visits can be considered for modeling the progression of rare disease.

Acknowledgement

This work was in part supported by the National Science Foundation award IIS-1418511, CCF-1533768 and IIS-1838042, the National Institute of Health award NIH R01 1R01NS107291-01 and R56HL138415.

References

Baytas, I. M.; Xiao, C.; Zhang, X.; Wang, F.; Jain, A. K.; and Zhou, J. 2017. Patient subtyping via time-aware lstm networks. In *KDD*, 65–74.

Bekker, J., and Davis, J. 2018. Learning from positive and unlabeled data: A survey. *arXiv preprint arXiv:1811.04820*.

Chawla, N. V.; Bowyer, K. W.; Hall, L. O.; and Kegelmeyer, W. P. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16:321–357.

Che, C.; Xiao, C.; Liang, J.; Jin, B.; Zho, J.; and Wang, F. 2017a. *An RNN Architecture with Dynamic Temporal Matching for Personalized Predictions of Parkinson's Disease*. 198–206.

Che, Z.; Cheng, Y.; Zhai, S.; Sun, Z.; and Liu, Y. 2017b. Boosting deep learning risk prediction with generative adversarial networks for electronic health records. In *ICDM*, 787–792. IEEE.

Choi, E.; Bahadori, M. T.; Sun, J.; Kulas, J.; Schuetz, A.; and Stewart, W. 2016. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In *NeurIPS*, 3504–3512.

Choi, E.; Biswal, S.; Malin, B.; Duke, J.; Stewart, W. F.; and Sun, J. 2017. Generating multi-label discrete patient records using generative adversarial networks. In *MLHC*, 286–305.

Choi, E.; Xiao, C.; Stewart, W.; and Sun, J. 2018. Mime: Multilevel medical embedding of electronic health records for predictive healthcare. In *NeurIPS*, 4547–4557.

Claesen, M.; De Smet, F.; Gillard, P.; Mathieu, C.; and De Moor, B. 2015. Building classifiers to predict the start of glucose-lowering pharmacotherapy using belgian health expenditure data. *arXiv preprint arXiv:1504.07389*.

Cordeiro, C. R.; Alfaro, T. M.; and Freitas, S. 2013. Clinical case: Differential diagnosis of idiopathic pulmonary fibrosis. *BMC research notes* 6(1):S1.

Dai, B., and Hu, B. 2010. Minimum conditional entropy clustering: A discriminative framework for clustering. In *ACML*, 47–62.

Deng, Y.; Shen, Y.; Jin, H.; et al. 2017. Disguise adversarial networks for click-through rate prediction. In *IJCAI*, 1589–1595.

Elkan, C., and Noto, K. 2008. Learning classifiers from only positive and unlabeled data. In *KDD*, 213–220.

Fung, G. P. C.; Yu, J. X.; Lu, H.; and Yu, P. S. 2005. Text classification without negative examples revisit. *TKDE* 18(1):6–20.

Galárraga, L.; Teflioudi, C.; Hose, K.; and Suchanek, F. M. 2015. Fast rule mining in ontological knowledge bases with amie+. *The VLDB Journal* 24(6):707–730.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NeurIPS*. 2672–2680.

Hinton, G.; Srivastava, N.; and Swersky, K. 2012. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. 14:8.

Kiryo, R.; Niu, G.; du Plessis, M. C.; and Sugiyama, M. 2017. Positive-unlabeled learning with non-negative risk estimator. In *NeurIPS*, 1675–1685.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *ICCV*, 2980–2988.

Liu, B.; Dai, Y.; Li, X.; Lee, W. S.; and Philip, S. Y. 2003. Building text classifiers using positive and unlabeled examples. In *ICDM*, volume 3, 179–188. Citeseer.

Ma, F.; Chitta, R.; Zhou, J.; You, Q.; Sun, T.; and Gao, J. 2017. Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In *KDD*, 1903–1911. ACM.

Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *JMLR* 9(Nov):2579–2605.

McLachlan, S.; Dube, K.; and Gallagher, T. 2016. Using the caremap with health incidents statistics for generating the realistic synthetic electronic healthcare record. In *ICHI*, 439–448.

Meltzer, E., and Noble, P. 2008. Idiopathic pulmonary fibrosis. *Orphanet Journal of Rare Diseases*.

Sellamanickam, S.; Garg, P.; and Selvaraj, S. K. 2011. A pairwise ranking based approach to learning with positive and unlabeled examples. In *CIKM*, 663–672.

Shire, H. 2013. Rare disease impact report: Insights from patients and the medical community.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*, 5998–6008.

Vickers, P. 2019. Challenges and opportunities in the treatment of rare diseases. *Drug Discovery World*.

Xiao, C.; Choi, E.; and Sun, J. 2018. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. *Journal of the American Medical Informatics*.

Yu, K.; Wang, Y.; Cai, Y.; Xiao, C.; Zhao, E.; Glass, L.; and Sun, J. 2019. Rare disease detection by sequence modeling with generative adversarial networks. *arXiv preprint arXiv:1907.01022*.

Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; and Schölkopf, B. 2004. Learning with local and global consistency. In *NeurIPS*, 321–328.